TESIS – KI142502

Rekomendasi Kolaborasi Penelitian Antardomain Menggunakan Metode *Cross-Domain Topic Learning* Berbasis Frase

Vit Zuraida
5116201021

DOSEN PEMBIMBING
Dr. Eng. Chastine Fatichah, M.Kom
NIP: 19751220 20011220 02

PROGRAM MAGISTER
BIDANG KEAHLIAN KOMPUTASI CERDAS DAN VISI
DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN KOMUNIKASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2018

Tesis ini disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Komputer (M. Kom)
di

Institut Teknologi Sepuluh Nopember Surabaya

oleh:
VIT ZURAIDA
Nrp. 5116201021

Dengan judul:
Rekomendasi Kolaborasi Penelitian Antardomain Menggunakan Metode *Cross-Domain Topic Learning* Berbasis Frase
Tanggal Ujian : 27 Juli 2018
Periode Wisuda : 2018 Gasal

Disetujui oleh:

1. Dr. Eng. Chastine Fatichah, M.Kom. ..... (Pembimbing I)
   NIP. 197512202001122002
2. Prof.Dr.Ir.Joko Lianto Buliali, M.Sc ..... (Penguji I)
   NIP. 196707271992031002
3. Dr. Eng. Darlis Heru Murti, S.Kom, M.Kom ..... (Penguji II)
   NIP. 197712172003121001
4. Dr. Eng. Nanik Suciati, S.Kom, M.Kom ..... (Penguji III)
   NIP. 197104281994122001

Dekan Fakultas
Teknologi Informasi dan Komunikasi,

Dr. Agus Zainal Arifin, S.Kom, M.Kom
NIP. 19720809 199512 1 001

REKOMENDASI KOLABORASI PENELITIAN ANTARDOMAIN MENGGUNAKAN METODE *CROSS-DOMAIN TOPIC LEARNING* BERBASIS FRASE

Nama : Vit Zuraida

NRP : 5116201021

Pembimbing : Dr. Eng. Chastine Fatichah, S.Kom, M.Kom

ABSTRAK

Rekomendasi kolaborasi antardomain memiliki permasalahan terkait kelangkaan topik kolaborasi antardomain yang relevan sehingga berpengaruh terhadap akurasi rekomendasi. Sistem rekomendasi dibangun berdasarkan dokumen penelitian yang pernah dipublikasikan, baik judul, abstrak, bibliografi, maupun isi secara keseluruhan. Oleh karena itu, proses ekstraksi topik riset seorang peneliti merupakan tahapan penting. Model topik berbasis *bag-of-words* belum dapat merepresentasikan topik dengan baik sebab urutan kata pada dokumen tidak diperhitungkan. Penelitian ini mengusulkan sistem rekomendasi kolaborasi penelitian antardomain menggunakan metode *Cross-Domain Topic Learning* (CTL) Berbasis Frase yang memperhatikan urutan kata. CTL dengan frase juga mempertimbangkan kelangkaan topik kolaborasi antardomain.

Sistem rekomendasi kolaborasi yang diusulkan terdiri dari tiga fase utama. Fase pertama adalah transformasi dokumen dari format *bag-of-words* menjadi *bag-of-phrases.* Fase kedua adalah pemodelan topik terhadap frase yang sudah dibentuk. Hasilnya adalah distribusi probabilitas keterkaitan peneliti dengan topik. Nilai probabilitas tersebut selanjutnya dijadikan input dalam fase *Random Walk with Restart* yang menghasilkan rekomendasi kolaborator.

Uji coba dilakukan pada domain *visualization* dan *data mining* dari dataset penelitian AMiner untuk maksimal tiga kata dalam frase. Hasil uji coba menunjukkan bahwa rekomedasi yang dihasilkan CTL Berbasis Frase lebih baik daripada CTL berbasis kata tunggal (*bag-of-words*). Terdapat peningkatan nilai *precision* sebesar ±10% pada 10 rekomendasi teratas dan ±5% pada 20 rekomendasi teratas untuk kebenaran hasil rekomendasi.

Kata kunci: model topik, rekomendasi kolaborasi antardomain, *random walk*

[Halaman ini sengaja dikosongkan]

CROSS-DOMAIN RESEARCH COLLABORATION RECOMMENDATION USING PHRASE-BASED CROSS-DOMAIN TOPIC LEARNING

Student Name	: Vit Zuraida
Student Identity Number	: 5116201021
Supervisor	: Dr. Eng. Chastine Fatichah, S.Kom, M.Kom

ABSTRACT

Cross-domain collaboration have several specific issues including the rarity of relevant cross-domain collaboration topics that could affect the accuracy of the recommendation. Recommendation systems are built based on published research documents, including titles, abstracts, bibliographies, or the entire content of the documents. Therefore, the process of extracting research topics from a researcher is an important step. Topic modeling based on bag-of-words are not able to represent the topic effectively because the order of words in the document is not considered. This research proposes cros-domain research collaboration recommendation system using Phrase-Based Cross-Domain Topic Learning (CTL) method that considers word order. Phrase-Based CTL also considers the rarity of cross-domain collaboration topics.

Phrase-Based CTL consists of three main phases. The first phase is the transformation of documents from the bag-of-words format into bag-of-phrases. The second phase is the topic modeling of the established phrases. The result is probability distribution of the researcher's relevance to each topic. The probability distribution is then used as input in the Random Walk with Restart phase resulting in collaborator ranking.

Experiments were conducted on the domain visualization and data mining of the AMiner research dataset for a maximum of three words in a phrase. Experimental result shows that the recommendations produced by Phrase-Based CTL are better than CTL based on bag-of-words. There is ± 10% improvement of precision value in the top 10 recommendations and ± 5% improvement in the top 20 recommendations.

***Keywords*:** *topic model, cross-domain collaboration recommendation, random walk*

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Allah SWT yang telah melimpahkan kasih dan sayang-Nya kepada kita, sehingga penulis bisa menyelesaikan tesis dengan tepat waktu, dengan judul "**Rekomendasi Kolaborasi Penelitian Antardomain Menggunakan Metode *Cross-Domain Topic Learning* Berbasis Frase**". Tujuan dari penyusunan tesis ini guna memenuhi salah satu syarat untuk bisa menempuh ujian magister komputer di Fakultas Teknologi Informasi dan Komunikasi (FTIK) Program Studi S2 Informatika Institut Teknologi Sepuluh Nopember Surabaya (ITS).

Didalam pengerjaan tesis ini telah melibatkan banyak pihak yang sangat membantu dalam banyak hal. Oleh sebab itu, disini penulis sampaikan rasa terima kasih sedalam-dalamnya kepada:

1. Kedua Orang Tua dan keluarga yang selalu memberikan dukungan moril maupun materiil.
2. Ibu Chastine Fatichah dan Ibu Diana Purwitasari, selaku dosen pembimbing yang telah secara sabar memberikan ilmu dan waktunya kepada penulis untuk menyelesaikan tesis ini.
3. Bapak Joko Lianto Buliali, Bapak Darlis Heru Murti, dan Ibu Nanik Suciati selaku dewan penguji pada sidang tesis peneliti yang telah memberikan saran untuk perbaikan tesis ini.
4. Teman-Teman S2 Informatika 2016 yang tidak dapat disebutkan satu-persatu. Terima kasih atas bantuan dan semangat yang diberikan selama masa studi.

Penulis menyadari bahwa tesis ini masih jauh dari sempurna, karena kesempurnaan hanya milik Allah SWT. Oleh karena itu, masukan dan saran yang bersifat membangun sangat penulis harapkan. Akhirnya, penulis berharap agar tesis ini mampu memberikan kontribusi yang bermanfaat bagi bidang keilmuan di kemudian hari.

Surabaya, 30 Juni 2018

Vit Zuraida

[Halaman ini sengaja dikosongkan]

DAFTAR ISI

DAFTAR GAMBAR

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

[Halaman ini sengaja dikosongkan]

BAB 1
PENDAHULUAN

Pada bab ini akan dijelaskan mengenai beberapa hal dasar dalam pembuatan proposal penelitian yang meliputi latar belakang, perumusan masalah, tujuan, manfaat, dan batasan masalah.

1.1. Latar Belakang

Seiring dengan berkembangnya ilmu pengetahuan, berbagai jenis penelitian tidak hanya terkonsentrasi pada suatu domain tertentu, namun bisa jadi merupakan kolaborasi dari beberapa domain. Kolaborasi antardomain ini menggabungkan beragam keahlian serta metode dan telah terbukti efektif dalam pemecahan masalah kompleks yang akhirnya memunculkan hasil yang inovatif baik secara teoritis maupun aplikatif (Liang et al. 2017), misalnya kolaborasi penelitian dalam hal efisiensi energi (Hemptinne et al. 2017), teknik biomedis (Abed et al. 2017), dan konservasi alam (Mitchell et al. 2017).

Rekomendasi kolaborasi bertujuan untuk membantu peneliti dalam menemukan kolaborator baik pada domain yang sama, antardomain, antarinstitusi, maupun antara institusi dengan industri (Q. Wang et al. 2017). Secara garis besar sistem rekomendasi terbagi menjadi dua kategori, yaitu *collaborative filtering* yang dibangun berdasarkan similaritas *user*, dalam hal ini peneliti, dan *content-based* yang dibangun berdasarkan similaritas *item*, misalnya isi konten publikasi penelitian (Kang, Li, and Coppel 2015). Beberapa metode telah dikembangkan dalam identifikasi dan rekomendasi penelitian antardomain, di antaranya HyClass yang menggunakan similaritas semantik pada term dalam publikasi dengan memanfaatkan taksonomi MeSH (Kang, Li, and Coppel 2015). Osuna (2017) membentuk rekomendasi kolaborasi dengan melakukan klasifikasi peneliti berdasarkan *research footprint* (kata kunci, konsep, atau judul dan abstrak) yang dimiliki. Liang (2017) memberikan rekomendasi melalui metode *clustering* dengan cara menghitung nilai similaritas antartopik.

Tidak seperti kolaborasi pada domain yang sama, rekomendasi kolaborasi antardomain lebih sulit dibentuk. Jie Tang menyimpulkan hal ini disebabkan oleh tiga hal (Tang et al. 2012). Penyebab pertama adalah koneksi yang tersebar. Peneliti pada suatu domain seringkali tidak memiliki relasi dengan peneliti lain pada domain yang

berbeda sehingga akan sulit untuk menemukan kolaborator yang tepat. Permasalahan kedua adalah perbedaan keahlian yang seringkali menyebabkan perbedaan terminologi untuk kasus yang serupa pada domain lainnya. Masalah ketiga adalah relevansi topik untuk kolaborasi antardomain. Penelitian Tang menunjukkan bahwa ternyata hanya 9% dari seluruh kemungkinan pasangan domain yang diuji memiliki kolaborasi penelitian. Ketiga permasalahan tersebut mendorong Tang mengusulkan metode *Cross-Domain Topic Learning* (CTL) yang mempertimbangkan bahwa terdapat topik yang eksklusif terkait dengan satu domain saja sehingga tidak relevan terhadap kolaborasi antardomain. Metode ini bertujuan untuk memastikan bahwa kelangkaan topik yang relevan tidak memberikan pengaruh yang buruk pada akurasi sistem rekomendasi.

CTL termasuk model rekomendasi kolaborasi antardomain yang dibentuk berdasarkan similaritas konten (judul dan abstrak dari dokumen penelitian). Oleh karena itu, proses ekstraksi topik riset dari seorang peneliti merupakan tahapan yang penting. Model topik yang digunakan adalah *Latent Dirichlet Allocation* (LDA) dengan pendekatan *bag-of-words*. Hal ini akan memberikan dampak negatif pada rekomendasi yang dihasilkan sebab suatu kata bisa saja terkait dengan topik yang berbeda jika berada pada frase yang berbeda, misalnya kata "*model"* dalam frase '*mining model"* dan "*mathematical model"*. Model topik dengan pendekatan *bag-of-words* mengasumsikan bahwa setiap kata berdiri secara independen, padahal bahasa memiliki prinsip *non-compositionality*, yang berarti bahwa makna frase tidak selalu dapat disimpulkan dari makna masing-masing kata yang menyusunnya (Schone and Jurafsky 2001). Frase yang dibentuk dengan kombinasi kata dan urutan tertentu umumnya menyimpan informasi yang lebih penting dibandingkan dengan gabungan nilai informasi dari setiap kata di dalamnya (X. Wang, McCallum, and Wei 2007). Oleh karena itu, model topik berbasis frase dikembangkan dengan argumentasi bahwa frase lebih representatif dalam menentukan topik dibandingkan dengan kata secara individual, misalnya frase "*support vector machine"* merepresentasikan topik *machine learning* lebih baik dibandingkan kata "*support*", "*vector*", dan "*machine*" secara terpisah. Beberapa metode model topik berbasis frase yang telah dikembangkan antara lain, KERT (Danilevsky et al. 2013), TurboTopic (Blei and Lafferty 2009), *Phrase-Discovering LDA* (Lindsey, III, and Stipicevic 2012), dan ToPMine (El-Kishky et al. 2014).

Penelitian ini mengusulkan metode rekomendasi kolaborasi penelitian antardomain dengan metode *Cross-Domain Topic Learning* Berbasis Frase. Metode rekomendasi dasar yang dipakai adalah *Cross-Domain Topic Learning*. CTL dipilih karena berbeda dengan metode lainnya, metode ini mempertimbangkan adanya topik yang tidak relevan terhadap kolaborasi antardomain sehingga kelangkaan topik yang relevan tidak berpengaruh buruk terhadap akurasi model. Pengembangan CTL dilakukan dengan melakukan pemodelan topik berdasarkan frase. Transformasi *bag-of-words* pada dokumen menjadi *bag-of-phrases* dilakukan dengan metode ToPMine. ToPMine juga merupakan pengembangan LDA sehingga integrasi dengan metode dasar CTL lebih mudah dilakukan. Di samping itu, ToPMine merupakan metode sederhana yang independen terhadap domain, mampu memfilter kandidat frase yang tidak tepat, serta memiliki skalabilitas dan *run time* yang lebih baik dibandingkan dengan LDA dan beberapa model topik berbasis frase yang disebutkan sebelumnya. Dengan hasil ekstraksi topik yang lebih representatif diharapkan sistem rekomendasi kolaborasi antardomain yang diusulkan memiliki performa yang lebih baik.

1.2. Perumusan Masalah

Rumusan masalah yang diangkat dalam penelitian ini adalah sebagai berikut:

1. Bagaimana proses transfomasi *bag-of-words* menjadi *bag-of-phrases* menggunakan metode ToPMine?
2. Bagaimana memperoleh rekomendasi kolaborasi antardomain dengan metode *Cross-Domain Topic Learning* dengan pendekatan *bag-of-phrases*?
3. Bagaimana perbandingan performa hasil rekomendasi kolaborasi antardomain dengan pendekatan *bag-of-words* dan *bag-of-phrases?*

1.3. Tujuan dan Manfaat Penelitian

Tujuan penelitian ini adalah membangun metode rekomendasi kolaborasi antardomain menggunakan metode *cross-domain topic learning* dengan pendekatan *bag-of-phrases*. Manfaat dari penelitian ini adalah dihasilkannya suatu sistem yang dapat memberikan rekomendasi kolaborator dari domain tertentu untuk seorang peneliti dari domain lainnya. Rekomendasi yang diberikan akan mempermudah peneliti dalam

menemukan kolaborator yang tepat, mengembangkan inovasi riset antardomain yang sudah ada, serta mendorong munculnya kolaborasi riset antardomain yang baru.

1.4. Kontribusi Penelitian

Kontribusi penelitian ini adalah pengembangan metode rekomendasi kolaborasi penelitian antardomain dari metode dasar *Cross-Domain Topic Learning* berbasis frase. Ekstraksi topik dengan pendekatan *bag-of-phrases* dengan metode ToPMine diharapkan dapat menanggulangi masalah identifikasi topik pada kata-kata yang bisa diasosiasikan dengan topik yang berbeda bergantung pada frase yang dibentuk oleh kata tersebut.

1.5. Batasan Masalah

Batasan masalah pada penelitian ini adalah:

1. Rekomendasi kolaborasi antardomain diperoleh berdasarkan ekstraksi model topik terhadap bagian judul dan abstrak dari suatu dokumen penelitian.
2. Dokumen penelitian yang dijadikan data uji dalam penelitian ini adalah dokumen penelitian berbahasa Inggris.

BAB 2
KAJIAN PUSTAKA

Pada bab ini akan dijelaskan mengenai dasar teori dan kajian pustaka yang terkait dengan penelitian ini. Pustaka yang terkait adalah mengenai model topik, termasuk ToPMine, metode rekomendasi dengan *Cross-Domain Topic Learning*, dan *Random Walk with Restart*.

2.1. Model Topik

Model topik merupakan metode dalam melakukan analisa topik pada data teks tak terstruktur, seperti *social media*, berita, maupun dokumen publikasi ilmiah. Model topik dengan pendekatan statistik memodelkan dokumen sebagai kumpulan dari beberapa topik, sementara topik dimodelkan sebagai distribusi dari kata-kata. Beberapa model topik seperti PLSA (*probabilistic latent semantic analysis*) dan LDA (*latent dirichlet allocation*) merupakan *generative model* yang mengasumsikan bahwa dokumen dihasilkan oleh proses tertentu (Han and Wang 2014). Untuk menghasilkan kata-kata pada dokumen, sebuah *latent topic* dipilih dari sekumpulan topik berdasarkan distribusi multinomial. Kemudian sampel kata diambil berdasarkan distribusi kata pada topik tersebut. Dengan asumsi ini, distribusi kata pada topik yang mulanya tidak diketahui dapat diperoleh sesuai dengan frekuensi kemunculan kata pada dokumen.

Model topik dengan pendekatan *bag-of-words* seperti LDA mengasumsikan bahwa setiap kata berdiri secara independen, padahal frase lebih representatif dalam menentukan topik dibandingkan dengan kata secara individual (Schone and Jurafsky 2001), misalnya frase "*management system"* dan "*database system*" mampu merepresentasikan topik lebih baik dibandingkan kata "*system*" yang berdisi sendiri. Oleh karena itu, berbagai model topik berbasis frase dikembangkan agar urutan kata dalam frase dapat dipertimbangkan. Model topik berbasis frase umumnya terbagi menjadi beberapa kategori (Han and Wang 2014). Kategori pertama adalah model yang berusaha memperoleh frase dan topiknya secara bersamaan, seperti Topical N-Gram (X. Wang, McCallum, and Wei 2007) dan PD-LDA (*Phrase Discovering*-LDA) (Lindsey, III, and Stipicevic 2012). Topical N-Gram merupakan model probabilistic yang menghasilkan kata secara berurutan dengan memperhatikan kata dan topik sebelumnya

sedangkan PD-LDA menghasilkan kata dengan mempertimbangkan sejumlah *m* kata sebelumnya dan topik pada frase terakhir. Kategori kedua adalah model yang memvisualisasikan topik dengan *n-gram* setelah *model inference* terhadap *bag-of-words* selesai, misalnya KERT (Danilevsky et al. 2013) dan TurboTopics (Blei and Lafferty 2009). Setelah LDA selesai, KERT menggabungkan *unigram* berurutan yang memiliki topik yang sama jika dianggap signifikan, sedangkan TurboTopics mengembangkan model topik LDA dengan menambahkan proses *frequent phrase mining* di bagian akhir. Kategori ketiga adalah model topik yang mengekstrak *frequent phrase* yang selanjutnya dijadikan *input* dalam proses pemodelan yang sama dengan *bag-of-words,* misalnya ToPMine (El-Kishky et al. 2014).

2.2. TopMine

ToPMine merupakan salah satu *phrase-based topic model* yang dirancang berdasarkan argumentasi bahwa frase lebih representatif dalam menentukan topik daripada setiap kata yang membentuknya secara individu. ToPMine terdiri dari dua fase utama, yaitu:

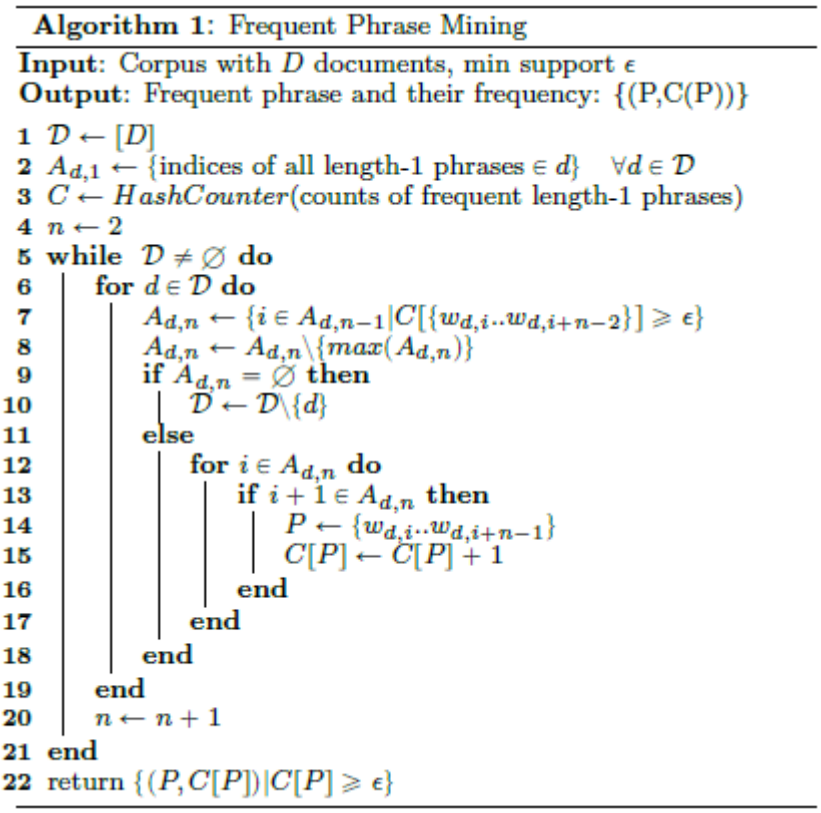
1. *Phrase Mining*

   Fase ini bertujuan untuk melakukan transformasi *bag-of-words* dari dokumen menjadi *bag-of-phrases* dalam dua tahapan berikut:

   a. *Frequent Phrase Mining*

      Tahapan ini bertujuan untuk memperoleh frase dengan frekuensi kemunculan lebih tinggi daripada *minsup* (*minimum support).* Fase ini memanfaatkan properti algoritma Apriori berikut:

      - *Downward closure lemma*: Jika suatu frase bukan merupakan *frequent phrase* maka semua *super-phrase* dari frase tersebut juga tidak termasuk *frequent phrase.*
      - *Data-antimonotocity*: Jika suatu dokumen tidak memiliki *frequent phrase* dengan panjang *n*, maka dokumen tersebut tidak mungkin memiliki *frequent phrase* dengan panjang lebih besar dari *n.*



Algorithm 1: Frequent Phrase Mining

Input: Corpus with D documents, min support ϵ
Output: Frequent phrase and their frequency: {(P,C(P))}

$\mathcal{D} \leftarrow [D]$
$A_{d,1} \leftarrow \{\text{indices of all length-1 phrases} \in d\} \quad \forall d \in \mathcal{D}$
$C \leftarrow HashCounter(\text{counts of frequent length-1 phrases})$
$n \leftarrow 2$
while $\mathcal{D} \neq \emptyset$ **do**
  for $d \in \mathcal{D}$ **do**
    $A_{d,n} \leftarrow \{i \in A_{d,n-1} | C[\{w_{d,i}..w_{d,i+n-2}\}] \geqslant \epsilon\}$
    $A_{d,n} \leftarrow A_{d,n} \backslash \{max(A_{d,n})\}$
    if $A_{d,n} = \emptyset$ **then**
      $\mathcal{D} \leftarrow \mathcal{D} \backslash \{d\}$
    else
      for $i \in A_{d,n}$ **do**
        if $i + 1 \in A_{d,n}$ **then**
          $P \leftarrow \{w_{d,i}..w_{d,i+n-1}\}$
          $C[P] \leftarrow C[P] + 1$
        end
      end
    end
  end
  $n \leftarrow n + 1$
end
return $\{(P, C[P]) | C[P] \geqslant \epsilon\}$

Gambar 2.1 Algoritma *Frequent Phrase Mining* (El-Kishky et al. 2014)

Algoritma yang digunakan dalam proses *frequent phrase mining* ditunjukkan pada Gambar 2.1. Masukan dari fase ini adalah korpus dokumen dan *minsup* yang nilainya ditentukan relatif terhadap ukuran korpus. Semakin tinggi *minsup* maka akan diperoleh nilai *precision* yang lebih tinggi dan nilai *recall* yang lebih rendah. Proses *phrase mining* dimulai dengan menyimpan indeks aktif dari posisi kata pada dokumen dengan frase yang panjangnya *n.* Selanjutnya, dilakukan penghitungan jumlah kemunculan frase pada setiap dokumen dalam korpus berdasarkan indeks aktif tersebut. Jika jumlah kemunculan tidak memenuhi *minsup*, maka sesuai dengan *downward closure lemma,* indeks aktifnya akan dieliminasi. Pada setiap iterasi juga diberlakukan *data-antimonotocity* sehingga jika suatu dokumen tidak memiliki *frequent phrase* dengan panjang tertentu, maka dokumen tersebut tidak diikutsertakan pada iterasi

berikutnya. Fase ini menghasilkan himpunan *frequent phrase* beserta frekuensinya.

b. *Phrase Construction*

Fase ini bertujuan untuk menemukan frase yang tepat dari keseluruhan hasil proses *frequent phrase mining.* Frase-frase tersebut difilter dengan pendekatan *bottom-up agglomerative merging.* Konstruksi frase dilakukan dengan menggabungkan rangkaian frase yang memiliki kemungkinan terbaik. Algoritma yang digunakan dalam proses frequent phrase mining ditunjukkan pada Gambar 2.2.

Algorithm 2: Bottom-up Construction of Phrases from Ordered Tokens

Input: Counter C, thresh α
Output: Partition

```
H ← MaxHeap()
Place all contiguous token pairs into H with their
  significance score key.
while H.size() > 1 do
    Best ← H.getMax()
    if Best.Sig ⩾ α then
        New ← Merge(Best)
        Remove Best from H
        Update significance for New with its left phrase
          instance and right phrase instance
    else
        break
    end
end
```

Gambar 2.2 Algoritma *Phrase Contruction* (El-Kishky et al. 2014)

Masukan dari fase ini adalah *frequent phrase* besera frekuensinya. Kemudian dilakukan penghitungan *significance score* untuk setiap pasang frase yang berurutan. *Significance score* frase P_1 dan P_2 diperoleh dengan formula 2.1. $f(P_1 \oplus P_2)$ merupakan frekuensi kemunculan rangkaian frase yang dibentuk P_1 dan P_2 sedangkan $\mu_0(P_1, P_2)$ adalah rata-rata frekuensi kemunculan rangkaian P_1 dan P_2 berdasarkan *null hypothesis* terhadap kemandirian frase P_1 dan P_2.

$$sig(P_1, P_2) \approx \frac{f(P_1 \oplus P_2) - \mu_0(P_1, P_2)}{\sqrt{f(P_1 \oplus P_2)}} \quad (2.1)$$

Nilai $\mu_0\,(P_1, P_2)$ dihitung dengan formula 2.2 dimana L adalah jumlah token pada korpus dan $p(P) \approx \frac{f(P)}{L}$ adalah estimasi nilai probabilitas kemunculan frase dalam korpus.

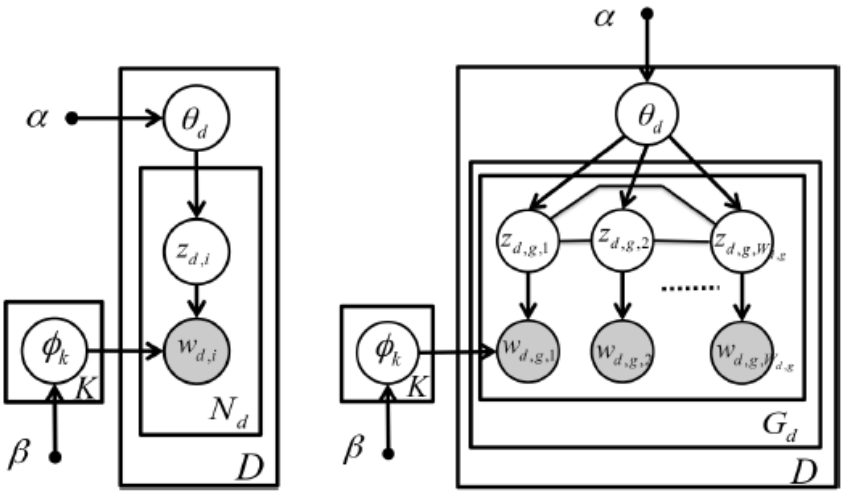
$$\mu_0\,(P_1, P_2) = Lp(P_1)p(P_2) \tag{2.2}$$

Nilai *significance score* terbaik selanjutnya menjadi acuan dalam menentukan pasangan frase yang paling tepat untuk digabung. Proses ini dilakukan hingga tidak ada *significance score* yang memenuhi *threshold* atau seluruh frase sudah digabungkan. Hasil akhir dari fase ini adalah *bag-of-words* dari dokumen pada korpus.

2. *PhraseLDA*

Pada tahap ini, *bag-of-phrases* yang dibentuk pada fase sebelumnya dijadikan dasar dalam ekstraksi topik seperti halnya pada algoritma *Latent Dirichlet Allocation*. Pada suatu frase (disebut juga *clique* $C_{d,g}$) yang memiliki sejumlah *s* kata, terdapat sebanyak K^s kemungkinan variasi topik. Namun, *PhraseLDA* memberlakukan *potential function* pada formula 2.3.

$$f(C_{d,g}) = \begin{cases} 1\ if\ z_{d,g,1} = z_{d,g,2} = \cdots = z_{d,g,|W_{d,g}|} \\ 0\ otherwise \end{cases} \tag{2.3}$$



Gambar 2.3 Representasi LDA (kiri) dan *PhraseLDA (kanan)* (El-Kishky et al. 2014)

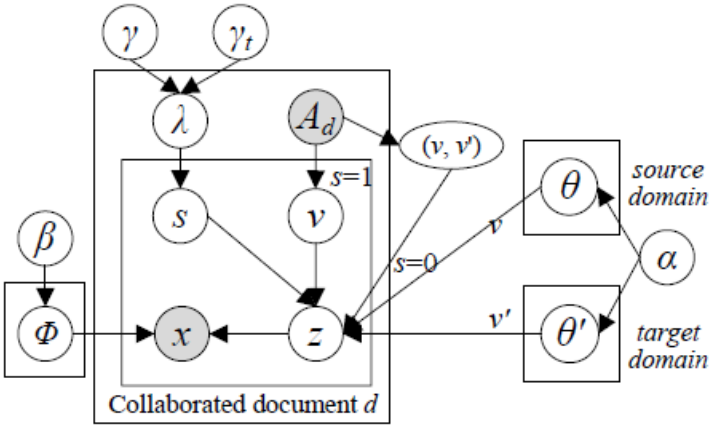
Potential function ini memaksa semua kata pada frase untuk dikaitkan dengan topik yang sama sehingga kemungkinan topik berkurang menjadi sejumlah *K*. Hal tersebut dimungkinkan karena adanya proses *agglomerative merging* dengan batasan *significance score* pada fase sebelumnya. Karakteristik inilah yang membedakan *PhraseLDA* dengan LDA seperti yang terlihat dalam Gambar 2.3. Dengan batasan ini pula, *Gibbs Sampling* pada *PhraseLDA* menggunakan formula 2.4 dalam penghitungan probabilitas posterior. $C_{d,g} = k$ digunakan untuk mengindikasikan bahwa semua kata pada frase dikaitkan dengan topik *k.* Notasi yang dipakai oleh *PhraseLDA* dirangkum dalam Tabel 2.1.

$$p(C_{d,g} = k \mid W, Z_{\setminus C_{d,g}}) \propto \prod_{j=1}^{W_{d,g}} (\alpha_k + \mathcal{N}_{d,k \setminus C_{d,g}} + j - 1) \frac{(\beta_{w_{d,g,j}} + \mathcal{N}_{w_{d,g,j},k \setminus C_{d,g}})}{(\sum_{x=1}^{V} \beta_x + \mathcal{N}_{k \setminus C_{d,g}} + j - 1)} \quad (2.4)$$

Tabel 2.1 Notasi *PhraseLDA* (El-Kishky et al. 2014)

Variabel	**Deskripsi**
D, K, V	Himpunan dokumen, topik, dan kamus kata
d, g, j, k, x	Indeks untuk dokumen, frase, token, topik, dan kata
N_d	Himpunan token pada dokumen ke-*d*
G_d	Himpunan frase pada dokumen ke-*d*
$W_{d,g}$	Himpunan token pada frase ke-*g* di dokumen ke-*d*
$z_{d,g,j}$	Topik dari token ke-*j* pada frase ke-*g* di dokumen ke-*d*
$w_{d,g,j}$	Token ke-*j* pada frase ke-*g* di dokumen ke-*d*
θ_d	Distribusi multionomial terhadap topik pada dokumen ke-*d*
ϕ_k	Distribusi multionomial terhadap kata pada topik *k*
$\mathcal{N}_k$	Jumlah token yang dikaitkan dengan topik *k*
$\mathcal{N}_{d,k}$	Jumlah token yang dikaitkan dengan topik *k* pada dokumen *d*
$\mathcal{N}_{x,k}$	Jumlah token yang nilainya *x* dan dikaitkan dengan topik *k*
α, β	Parameter Dirichlet untuk θ_d dan ϕ_k
$C_{d,g}$	$\{z_{d,g,j}\}_{j=1}^{\lvert W_{d,g} \rvert}$, yaitu koleksi topik pada frase ke-*g* di dokumen *d*

2.3. *Cross-Domain Topic Learning*

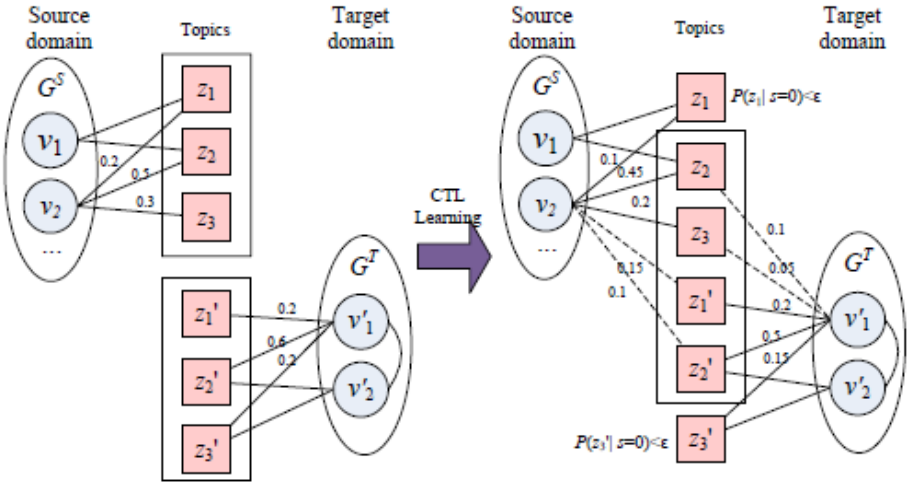


Gambar 2.4 Representasi Model CTL (Tang et al. 2012)

Hasil eksperimen menunjukkan bahwa hanya 9% dari kombinasi pasangan topik memiliki kolaborasi antardomain (Tang et al. 2012). Banyaknya topik yang tidak relevan terhadap kolaborasi antardomain ini bisa saja berdampak buruk terhadap performa model rekomendasi. Oleh karena itu, *Cross-Domain Topic Learning* (CTL) dibangun dengan mempertimbangkan topik yang tidak relevan terhadap kolaborasi antardomain. Model yang dibangun CTL direpresentasikan pada Gambar 2.4.

CTL membagi dokumen pada domain asal G^s dan domain target G^T menjadi tiga kategori, yaitu publikasi oleh penulis dari domain asal, publikasi oleh penulis dari domain target, dan publikasi yang merupakan kolaborasi antara kedua domain tersebut. Tahap pertama, CTL melakukan ekstraksi topik dari kedua domain (tanpa kategori kolaborasi antardomain) sesuai dengan distribusi $p(\theta_v|\alpha)$ dan $p(\theta'_{v'}|\alpha)$. Hasil dari tahap ini ditunjukkan pada Gambar 2.5 bagian kiri. Tahap kedua adalah pemodelan publikasi yang termasuk kategori kolaborasi antardomain. Untuk setiap x_{di}, nilai *s* diperoleh berdasarkan *beta distribution* $p(s|d) \sim beta(\gamma, \gamma_t)$. Jika *s* bernilai 1, seorang penulis *v* (atau *v'*) dipilih berdasarkan *uniform distribution.* Selanjutnya *sampling* dilakukan terhadap x_{di} dengan topik z_{di} spesifik terhadap user *v* sesuai dengan θ_v. Jika *s* bernilai 0, dipilih pasangan kolaborasi penulis (*v, v'*) dan distribusi multinomial $\vartheta_{vv'}$ dibentuk dengan menggabungkan θ_v dan $\theta_{v'}$. Penggabungan kedua model topik dilakukan dengan terlebih dahulu menyamakan dimensi keduanya. Selanjutnya adalah *sampling* x_{di} dari topik kolaborasi z_{di}berdasarkan distribusi $\vartheta_{vv'}$ yang baru. Hasil CTL *Learning* (Gambar 2.5 bagian kanan) memungkinkan penulis dari domain asal memiliki

distribusi terhadap topik pada domain target dan begitu juga sebaliknya. Notasi yang dipakai metode CTL dirangkum dalam Tabel 2.2.



Gambar 2.5 CTL *Learning* (Tang et al. 2012)

Tabel 2.2 Notasi CTL

Simbol	Deskripsi
T	Himpunan topik
d	Dokumen kolaborasi
A_d	Himpunan penulis untuk dokumen *d*
x_{di}	Token ke-*i* pada dokumen *d*
z_{di}	Topik yang dikaitkan dengan x_{di}
s_{di}	Bernilai 1 jika x_{di} adalah kata yang termasuk *single domain* atau dan 0 jika termasuk *cross domain*
θ dan θ'	Distribusi multinomial dari topik pada domain asal dan domain target
θ_v	Distribusi multinomial dari topik spesifik terhadap penulis *v*
$\vartheta_{vv'}$	Distribusi multinomial dari topik spesifik terhadap pasangan penulis (*v, v'*)
ϕ_z	Distribusi multinomial dari kata spesifik terhadap topik *z*
α, β	Parameter Dirichlet
λ	Parameter untuk *sampling* variabel *s*
γ, γ_t	Parameter beta untuk menghasilkan nilai λ

CTL menggunakan *Gibbs Sampling* untuk mengestimasi nilai parameter $\{\theta, \theta', \vartheta, \phi, \lambda\}$.

1. Probabilitas posterior pada *z* (atau *z'*) untuk setiap kata pada dokumen publikasi oleh penulis dari *single* domain menggunakan formula 2.5. Hasil perhitungan ini akan mempengaruhi nilai θ (atau θ').

$$P(z_{di}|z_{-di}, x, .) = \frac{n_{vz_{di}}^{-di} + \alpha}{\sum_z (n_{vz}^{-di} + \alpha)} \times \frac{m_{z_{di}x_{di}}^{-di} + \beta}{\sum_x (m_{z_{di}x}^{-di} + \beta)} \qquad (2.5)$$

   n_{vz} : Jumlah berapa kali topik *z* menjadi sampel dari distribusi multinomial spesifik terhadap penulis *v*

   m_{zx} : Jumlah berapa kali kata *x* dikaitkan dengan topik *z*

   n^{-di}: Notasi *–di* berarti jumlah tidak memperhitungkan item yang saat ini diproses

2. Probabilitas posterior pada *s* dihitung dengan formula 2.6 dan hasilnya digunakan untuk memperoleh nilai parameter $\theta, \theta', \vartheta$ dengan formula 2.7. Selanjutnya nilai ϕ, λ dapat disimpulkan dari model topik yang dibentuk. Formula 2.6 bisa disesuaikan untuk $P(s_{di} = 1|.)$. Perubahan yang perlu diperhatikan adalah mengganti $(n_{vz_{di}} + n_{v'z_{di}})$ dengan penulis tunggal $n_{vz_{di}}$ atau $n_{v'z_{di}}$.

$$P(s_{di} = 0|s_{-di}, z, .) = \frac{n_{ds_0}^{-di} + \gamma_t}{n_{ds_0}^{-di} + n_{ds_1}^{-di} + \gamma_t + \gamma} \times \frac{n_{vv'z_{di}}^{-di} + (n_{vz_{di}} + n_{v'z_{di}}) + \alpha}{\sum_z (n_{vv'z}^{-di} + (n_{vz_{di}} + n_{v'z_{di}}) + \alpha)} \qquad (2.6)$$

n_{ds_0} : Jumlah berapa kali 0 menjadi sampel pada dokumen *d*

(v, v'): Pasangan penulis yang dipilih untuk suatu x_{di}

$n_{vv'z}$: Jumlah berapa kali topik *z* dijadikan sampel untuk (v, v')

Probabilitas posterior topik *z* didefinisikan pada formula 2.7 sebagai berikut:

$$P(z_{di}|s_{di}=0,x,z_{-di},.)=\frac{m_{z_{di}x_{di}}^{-di}+m_{z_{di}x_{di}}+m'_{z_{di}x_{di}}+\beta}{\sum_x\left(m_{z_{di}x}^{-di}+m_{z_{di}x}+m'_{z_{di}x}+\beta\right)} \times \frac{n_{vv'z_{di}}^{-di}+\left(n_{vz_{di}}+n_{v'z_{di}}\right)+\alpha}{\sum_z(n_{vv'z}^{-di}+(n_{vz}+n_{v'z})+\alpha)} \quad (2.7)$$

m_{zx}^{-di}: Jumlah berapa kali kata *x* dikaitkan dengan topik *z* di publikasi kolaborasi

m_{zx} : Jumlah berapa kali kata *x* dikaitkan dengan topik *z* di publikasi domain asal

m'_{zx} : Jumlah berapa kali kata *x* dikaitkan dengan topik *z* di publikasi domain target

Pada proses estimasi nilai parameter, CTL menyimpan data berupa matriks jumlah *V x T* (penulis-topik), *D x 2* (dokumen-koin nilai *s*), *2 x T* (koin-topik), dan *AP x T* (pasangan penulis-topik). Hasil CTL *learning* kemudian dibangun menjadi sebuah *graph.* Suatu *node* penulis dihubungkan dengan suatu *node* topik jika memiliki probabilitas posterior lebih besar dari *threshold* ϵ $P(z|s=0,.)>\epsilon$ (tanda baca titik menunjukkan semua parameter harus dipertimbangkan dalam menghitung probabilitas). Semakin kecil *threshold* yang diberlakukan maka *graph* yang terbentuk akan semakin padat. *Random Walk with Restart* dijalankan untuk mengukur tingkat keterkaitan antara *node* penulis pada domain asal dengan *node* penulis pada domain target. *Node-node* dengan tingkat keterkaitan terbaik akan dijadikan rekomendasi kolaborasi antardomain.

2.4. *Random Walk with Restart*

Random Walk with Restart (RWR) telah banyak digunakan dalam berbagai aplikasi seperti *automatic image captioning, recommender system,* dan *link prediction* karena cukup efektif dalam mengukur tingkat keterkaitan yang baik antara dua node pada suatu *graph.* Beberapa kelebihan RWR adalah kemampuannya dalam menangkap multiple relasi antara dua *node* dan struktur *graph* secara global (Fujiwara et al. 2012). RWR dijalankan dengan berpindah dari suatu *node* ke *node* lain yang terhubung dengannya. Pada setiap langkah, terdapat kemungkinan τ bahwa perjalanan akan kembali ke *node* awal. r^t adalah vektor kolom dengan elemen r_u^t berisi probabilitas

random walk sampai di node *u* pada step *t*. *q* merupakan vektor kolom yang terdiri dari nilai 0 kecuali pada posisi elemen yang bersesuaian dengan *node* awal bernilai 1. *S* adalah *adjacency matrix* dari *graph* dengan elemen A_{uv} merupakan probabilitas perpindahan dari node *u* ke *v*. *Stationary probability* untuk setiap node dapat diperoleh dengan mengeksekusi formula 2.8 secara berulang hingga konvergensi dicapai.

$$r^{(t+1)} = (1-\tau)S.r^t + \tau q \tag{2.8}$$

[Halaman ini sengaja dikosongkan]

BAB 3
METODOLOGI PENELITIAN

Bab ini memaparkan metodologi penelitian yang digunakan dalam penelitian, meliputi (1) Studi Literatur, (2) Data Set, (3) Desain Model Sistem, (4) Skenario Uji Coba, (5) Analisa Hasil, dan (6) Penulisan Laporan.

3.1. Studi Literatur

Tahap awal dalam penelitian adalah studi literatur terkait topik penelitian yang dikaji. Studi literatur bertujuan menggali informasi mengenai dasar teori maupun perkembangan metode-metode yang sudah ada sebelumnya. Hasil studi literatur diharapkan dapat menjadi dasar dari metode yang diusulkan. Beberapa referensi yang dibutuhkan terkait dengan penelitian ini adalah mengenai model topik, ToPMine, *Cross-Domain Topic Learning*, dan *Random Walk with Restart.*

3.2. Data Set

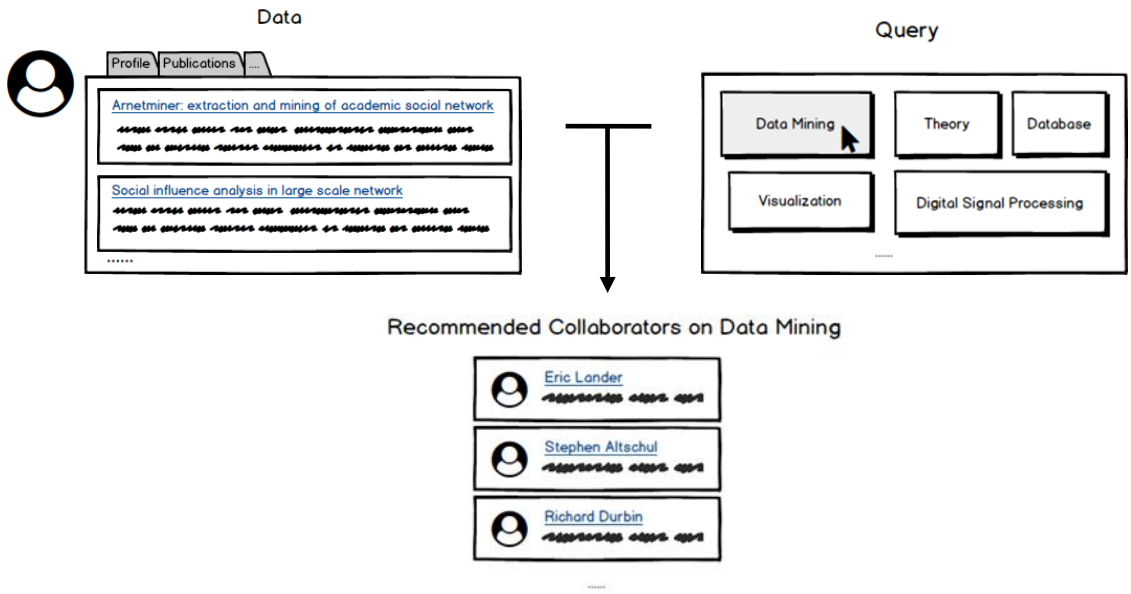
Publikasi ilmiah yang yang dijadikan sebagai data set dalam penelitian ini adalah abstrak, judul, dan peneliti dari *Arnetminer.org*. Domain yang dipilih untuk domain *Visualization* dan *Data Mining* dengan daftar jurnal maupun konferensi ditunjukkan pada Tabel 3.1. Data set dibagi menjadi dua bagian. Artikel ilmiah yang dipublikasikan pada tahun 2000 dan sebelumnya dijadikan sebagai data latih dan artikel ilmiah yang dipublikasikan setelah tahun 2000 dijadikan data uji.

Tabel 3.1 Data Uji

Domain	Jurnal/Konferensi	Jumlah Publikasi
Visualization	CVPR ICCV VAST TVCG IEEE *Visualization and Information Visualization*	3862
Data Mining	KDD SDM ICDM WSDM PKDD	2190

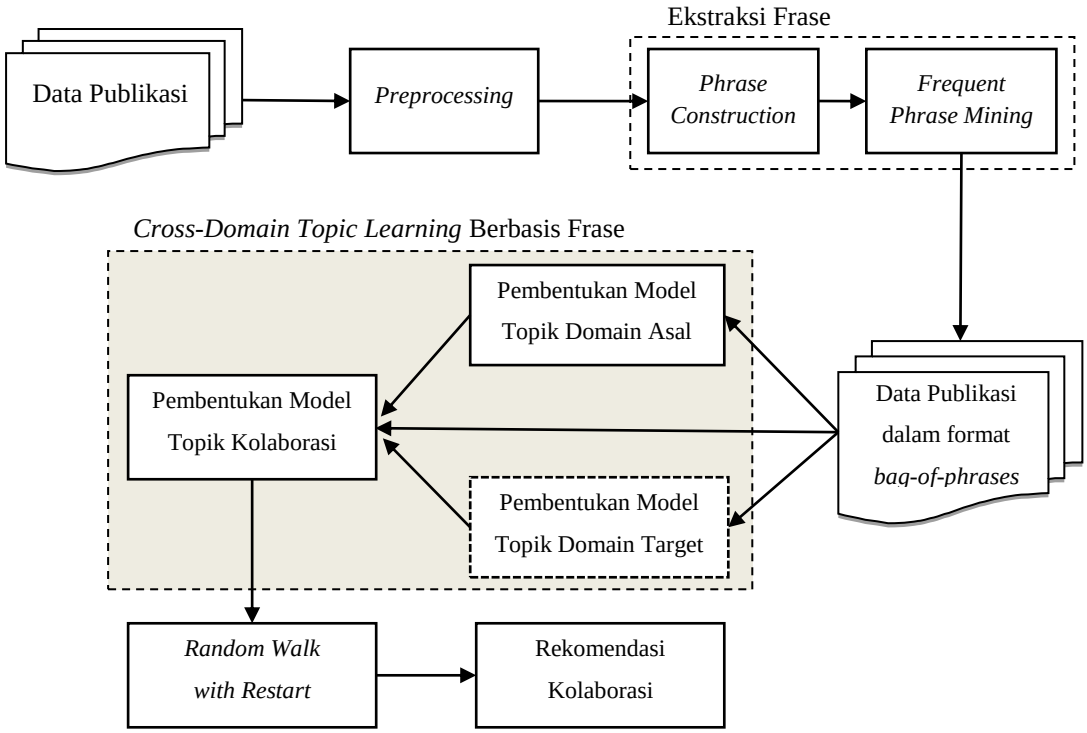
3.3. Desain Model Sistem

Sistem rekomendasi yang dibangun bertujuan untuk memberikan rekomendasi kolaborator dari domain tertentu untuk seorang peneliti dari domain lainnya berdasarkan data judul dan abstrak dari publikasi yang dimiliki. Misalnya peneliti dari domain *visualization* ingin menemukan kolaborator yang tepat pada domain *data mining*. Pada kasus tersebut, *visualization* merupakan domain asal dan *data mining* merupakan domain target. Seluruh artikel ilmiah yang dipublikasikan pada domain asal selanjutnya disebut publikasi domain asal dan artikel ilmiah yang dipublikasikan pada domain target disebut publikasi domain target. Seorang peneliti termasuk pada domain tertentu jika memiliki publikasi pada domain tersebut. Gambar 3.1 mengilustrasikan bahwa dengan data berupa judul dan abstrak publikasi, seorang peneliti bisa memilih domain target yang diinginkan dan sistem akan memberikan rekomendasi kolaborator yang sesuai dari domain tersebut.



Gambar 3.1 Ilustrasi Sistem Rekomendasi Kolaborasi Penelitian Antardomain

Penelitian ini terdiri dari dua tahapan utama. Tahap pertama adalah proses *training* untuk menghasilkan rekomendasi kolaborator berdasarkan data latih. Proses *training* dilakukan dengan membentuk sistem rekomendasi kolaborasi antardomain dalam empat fase yang ditunjukkan pada Gambar 3.2. Tahap kedua adalah proses *testing* yang dilakukan dengan cara menganalisa perbandingan hasil rekomendasi pada proses *training* terhadap *ground truth* pada data uji.



Gambar 3.2 Desain Model Sistem Rekomendasi Kolaborasi Antardomain

Pembentukan sistem rekomendasi kolaborasi antardomain terdiri dari empat fase utama. Fase pertama adalah tahap *preprocessing* terhadap judul dan abstrak setiap publikasi. Fase kedua adalah proses ekstraksi frase untuk melakukan transformasi *bag-of-words* menjadi *bag-of-phrases* menggunakan metode ekstraksi frase dari ToPMine. Pada fase ketiga, CTL Berbasis Frase digunakan untuk melakukan pemodelan topik dari setiap peneliti berdasarkan *bag-of-phrases* yang telah terbentuk. Topik didefinisikan sebagai bidang riset dengan lingkup yang lebih spesifik pada setiap domain, misalnya topik *association rule mining* dan *query processing* dalam domain data mining. Namun pada penelitian ini, model topik yang dihasilkan terbatas pada nilai probabilitas keterkaitan peneliti pada suatu topik tanpa ada pelabelan topik lebih lanjut. Pada tahap akhir, algoritma *Random Walk with Restart* diterapkan untuk memperoleh rekomendasi kolaborator pada domain target untuk seorang peneliti pada domain asal.

3.3.1. Fase *Preprocessing*

Pada fase ini dilakukan beberapa tahap *preprocessing*, yaitu:

1. *Case folding*, yaitu pengubahan keseluruhan teks input ke dalam dalam format yang sama, umumnya huruf kecil.

2. Tokenisasi, yaitu pemenggalan dokumen menjadi bagian-bagian tertentu yang disebut dengan token, dalam hal ini kata, dengan pemisah berupa spasi maupun tanda baca.
3. *Stopword removal,* yaitu proses eliminasi kata berfrekuensi tinggi pada dokumen, misalnya "*the*", "*and*", "*to*", dsb. Kata yang sering muncul umumnya tidak dapat menjadi pembeda antara satu dokumen dengan dokumen yang lain.
4. *Stemming,* yaitu proses konversi token menjadi akar katanya (*root).* Hal ini dilakukan dengan menghilangkan imbuhan pada kata turunan agar kembali ke kata dasar. *Stemmer* yang akan digunakan dalam penelitian ini adalah *Porter Stemmer.*

Contoh hasil fase *preprocessing* terhadap judul dan abstrak pada publikasi (Pay, 2016) ditunjukkan pada Gambar 3.3. Setiap token yang bukan *stopword* telah dihilangkan imbuhannya dan dikonversi menjadi huruf kecil.

Totally Automated Keyword Extraction

We develop and analyze an unsupervised and domain-independent method for extracting keywords from single documents. Our approach differs from the previous ones in the way of identifying candidate keywords, pruning the list of candidate keywords with several filtering heuristics and selecting keywords from the list of candidate keywords according to dynamic threshold functions. We were able to obtain better precision and recall on the same data set with our method than the previously studied ones.

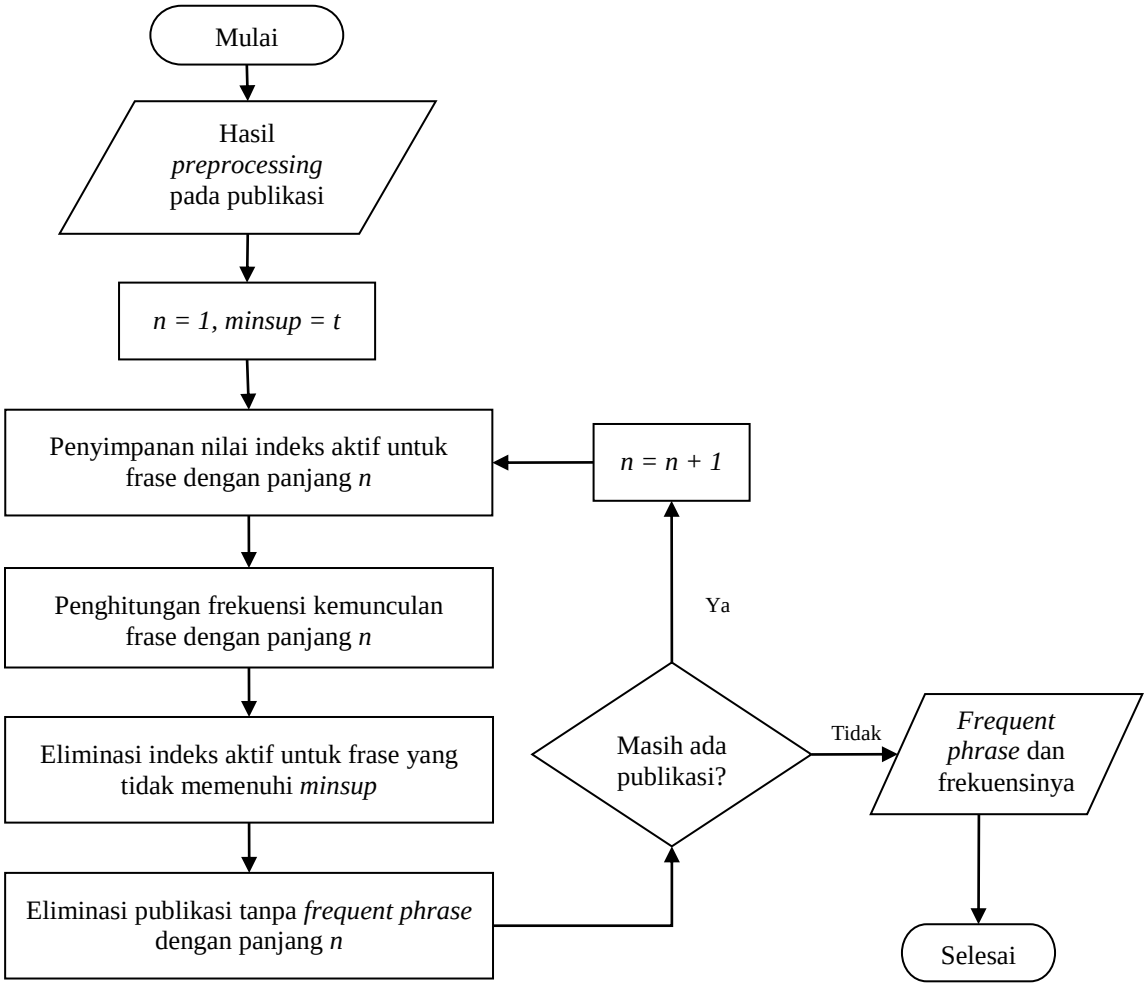
total autom keyword extract develop analyz unsupervis domain independ method extract keyword document approach differ identifi candid keyword prune candid keyword filter heurist select keyword candid keyword dynam threshold function precis recal data set method studi

Gambar 3.3 Contoh Hasil Fase *Preprocessing*

3.3.2. Fase Ekstraksi Frase

1. *Frequent Phrase Mining*

   Diagram alur proses ektraksi *frequent phrase* ditunjukkan pada Gambar 3.4. Masukan dari fase ini adalah publikasi yang sudah melalui proses *preprocessing* dan *minsup* (*minimum support*) yang nilainya ditentukan secara relatif terhadap ukuran korpus. Semakin tinggi *minsup* maka akan diperoleh nilai *precision* yang lebih tinggi dan nilai *recall* yang lebih rendah. Nilai minimum support yang optimal akan ditentukan pada tahap uji coba.



Gambar 3.4 Diagram Fase *Frequent Phrase Mining*

Adapun langkah-langkahnya dijelaskan sebagai berikut:

Langkah 1: Penyimpanan nilai indeks aktif untuk frase dengan panjang *n*. Pada iterasi pertama, indeks aktif adalah posisi kata pada dokumen setelah tahap *preprocessing*.

Langkah 2: Penghitungan frekuensi kemunculan setiap frase dengan panjang *n.*

Langkah 3: Eliminasi frase dengan frekuensi yang tidak memenuhi *minimum support* dengan cara menghapus indeks frase tersebut dari himpunan indeks aktif agar tidak diperhitungkan pada iterasi selanjutnya (*downward closure lemma*). Pada langkah ini juga dilakukan eliminasi indeks aktif terakhir pada masing-masing dokumen sebab frase yang lebih panjang tidak mungkin dapat dibentuk dari frase terakhir.

Sementara itu, frase yang memenuhi *minimum support* ditambahkan pada himpunan *frequent phrase P*.

Langkah 4: Eliminasi dokumen publikasi yang tidak memiliki *frequent phrase* dengan panjang *n*. Menurut *data-antimonotocity lemma* jika suatu dokumen tidak memiliki *frequent phrase* dengan panjang *n*, maka dokumen tersebut pasti tidak memiliki *frequent phrase* dengan panjang lebih dari *n*.

Langkah-langkah yang sama dilakukan untuk frase dengan panjang (n + 1) hingga tidak ada lagi dokumen yang dapat diproses. Hasil dari tahapan ini adalah daftar *frequent phrase* beserta frekuensinya. Berdasarkan contoh sebelumnya, gambaran hasil proses *frequent phrase* ditunjukkan dalam Tabel 3.2.

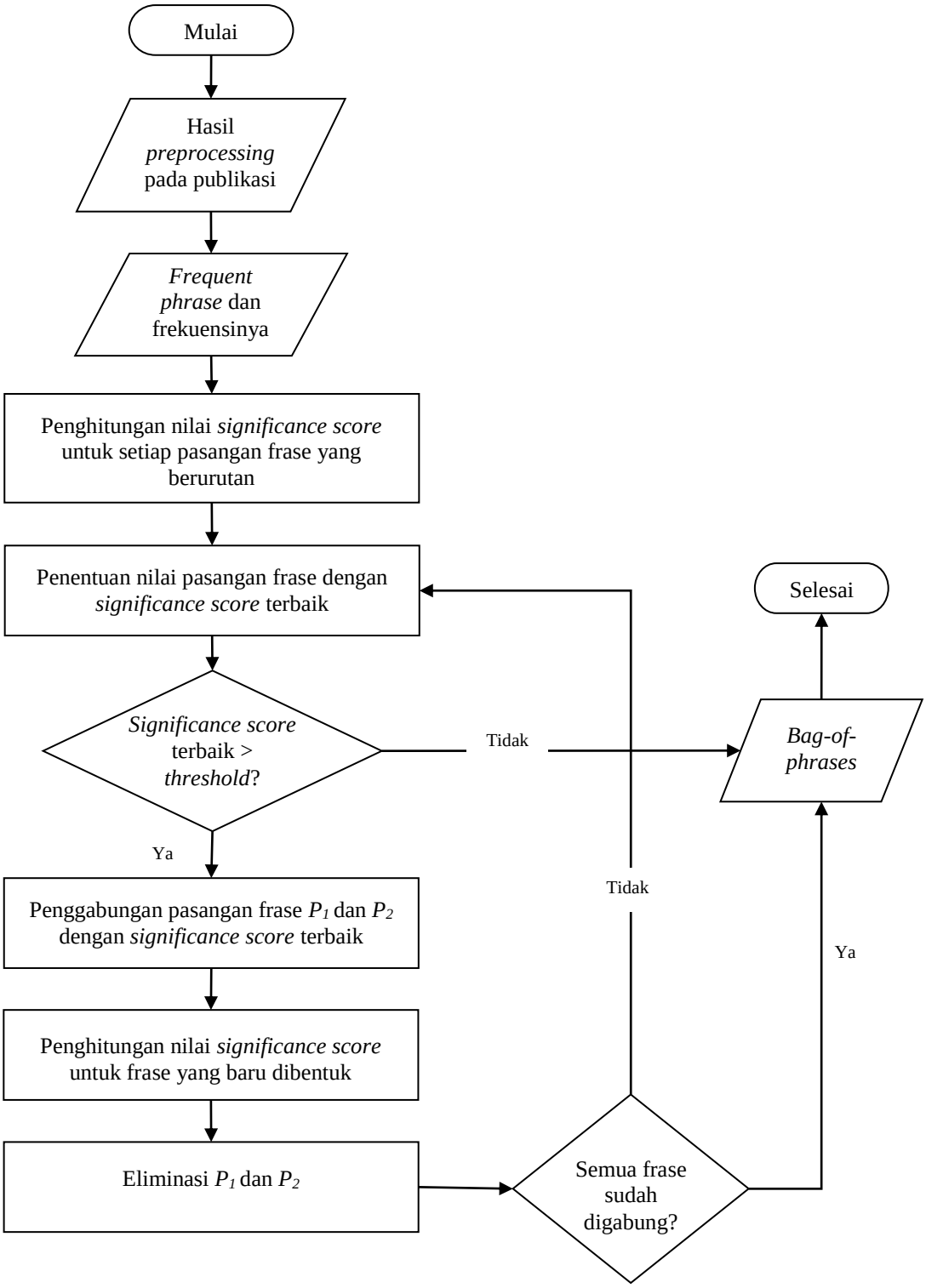
Tabel 3.2 Gambaran Hasil Proses *Frequent Phrase Mining*

Frase dengan 1 kata	**Frase dengan 2 kata**	**Frase dengan 3 kata**
total (21)	*autom candid (17)*	*total autom keyword (10)*
autom (10)	*candid keyword (25)*	*autom keyword extract (12)*
candid (20)	*total autom (15)*	*autom candid keyword (5)*
keyword (70)	*autom keyword (24)*	*domain indepen method (10)*
extract (51)	*keyword extract (35)*	*dynam threshold function (7)*
unsupervis (42)	*domain indepen (20)*	
domain (35)	*indepen method (12)*	
independ (26)	*dynam threshold (15)*	
method (29)	*threshold function (24)*	
dynam (28)		
threshold (32)		
function (40)		

2. *Phrase Construction*

Pada fase ini dilakukan konstruksi frase berdasarkan statistik *frequent phrase* yang telah diperoleh sebelumnya dengan metode *bottom-up agglomerative merging*. Proses ini bertujuan untuk memperoleh frase yang valid dan memfilter frase yang memenuhi *minimum support threshold* secara kebetulan. Hasil dari fase *phrase construction* adalah transformasi publikasi ke dalam bentuk *bag-of-phrases*. Alur fase ini ditunjukkan oleh Gambar 3.5 dengan penjelasan sebagai berikut:

Langkah 1: Penghitungan nilai *significance score*. Untuk setiap pasang frase P_1 dan P_2 yang berurutan, nilai *significance score* diperoleh berdasarkan formula 3.1 dengan $f(P_1 \oplus P_2)$ adalah frekuensi kemunculan rangkaian frase yang dibentuk P_1 dan P_2 sedangkan $\mu_0\ (P_1, P_2)$ adalah



Gambar 3.5 Diagram Fase *Phrase Construction*

rata-rata frekuensi kemunculan rangkaian P_1 dan P_2 berdasarkan *null hypothesis* terhadap kemandirian frase P_1 dan P_2.

$$sig(P_1, P_2) \approx \frac{f(P_1 \oplus P_2) - \mu_0 (P_1, P_2)}{\sqrt{f(P_1 \oplus P_2)}} \quad (3.1)$$

Nilai $\mu_0 (P_1, P_2)$ dihitung dengan formula 3.2 dimana L adalah jumlah token pada korpus dan $p(P) = \frac{f(P)}{L}$ adalah estimasi nilai probabilitas kemunculan frase dalam korpus.

$$\mu_0 (P_1, P_2) = Lp(P_1)p(P_2) \quad (3.2)$$

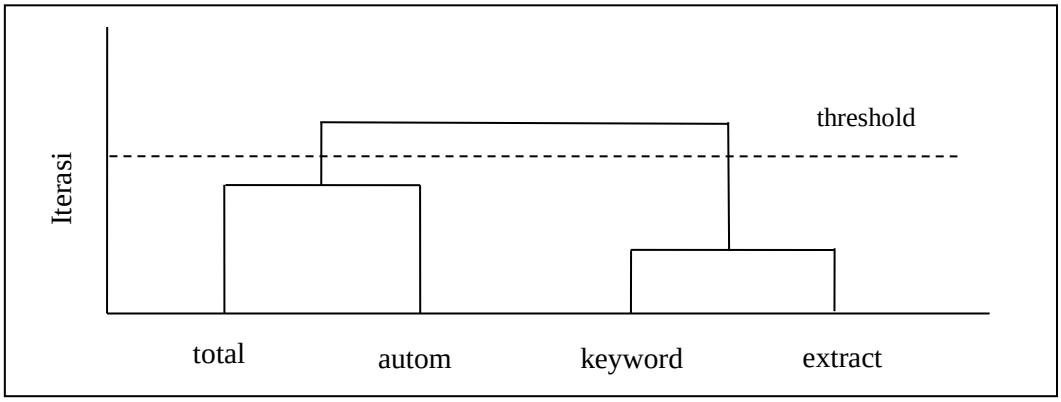
Langkah 2: Penentuan nilai pasangan frase dengan nilai *significance score* terbaik dari seluruh pasangan frase.

Langkah 3: Jika nilai *significance score* terbaik lebih besar dari *threshold*, maka pasangan frase tersebut digabungkan menjadi frase baru. Sebaliknya, jika nilai *significance score* lebih kecil daripada *threshold*, maka proses *phrase construction* dihentikan. Nilai *threshold* yang optimal untuk membentuk frase yang valid akan ditentukan pada tahap uji coba.

Langkah 4: Penghitungan nilai *significance score* untuk frase baru

Langkah 5: Eliminasi *node* frase P_1 dan P_2

Iterasi dalam fase *phrase construction* terus dilakukan hingga nilai *significance score* terbaik tidak memenuhi *threshold* atau ketika semua frase telah digabungkan.



Gambar 3.6 Contoh Proses *Phrase Construction*

Berdasarkan hasil proses *frequent phrase mining*, frase "*total autom keyword"* dan *"autom keyword extract"* termasuk *frequent phrase* karena kemunculannya memenuhi *minimum support.* Namun pada tahap *phrase*

construction, frase valid yang diperoleh berdasarkan nilai *significance score* adalah *"total autom"* dan *"keyword extract".* Gambaran hasil fase *phrase construction* ditunjukkan pada **Error! Not a valid bookmark self-reference.**.

Tabel 3.3 Gambaran Hasil Fase *Phrase Construction*

Bag-of-phrases **hasil *phrase construction***	
total autom	*prune*
keyword extract	*candid keyword*
develop	*filter*
analyz	*heurist select*
unsupervis	*keyword*
domain independ	*candid keyword*
method	*dynam threshold*
extract	*function*
keyword	*precis*
document	*recal*
approach	*data set*
differ	*method*
identifi	*studi*
candid keyword	

3.3.3. Fase *Cross-Domain Topic Learning* Berbasis Frase

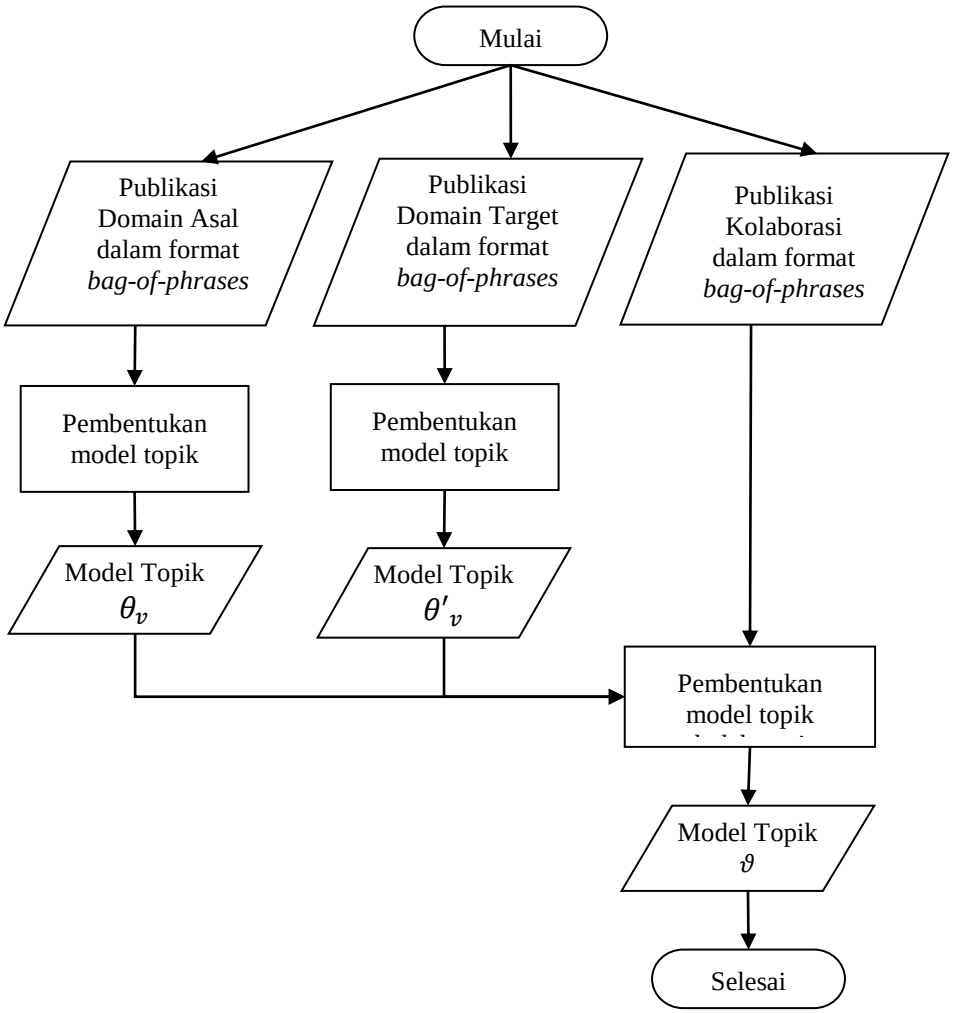
Hasil transformasi *bag-of-words* pada judul dan abstrak masing-masing publikasi menjadi *bag-of-phrases* selanjutnya dijadikan objek dalam pembentukan model topik untuk setiap peneliti dan pasangan peneliti dari domain asal dan domain target. Masukan pada fase ini adalah judul dan abstrak dari publikasi ilmiah yang sudah ditransformasikan ke dalam format *bag-of-phrases*.

Alur secara keseluruhan ditunjukkan oleh Gambar 3.7. Publikasi dibagi menjadi tiga kategori, yaitu publikasi dari domain asal, publikasi dari domain target, dan publikasi yang merupakan kolaborasi dari kedua domain tersebut. Untuk tahap awal, dilakukan pembentukan model topik pada domain asal dan domain target, sesuai dengan distribusi $p(\theta_v|\alpha)$ dan $p(\theta'_{v'}|\alpha)$. Tahap selanjutnya adalah pemodelan publikasi yang termasuk kategori kolaborasi antardomain. Untuk setiap C_{dg}, nilai s diperoleh berdasarkan *beta distribution* $p(s|d) \sim beta(\gamma, \gamma_t)$. Jika s bernilai 1 yang berarti publikasi pada satu domain, maka penulis v (atau v') dipilih berdasarkan *uniform distribution.* Selanjutnya *sampling* dilakukan terhadap C_{dg} dengan topik z_{dg} spesifik terhadap user v sesuai dengan θ_v. Jika s bernilai 0, dipilih pasangan kolaborasi penulis (v, v') dan distribusi multinomial $\vartheta_{vv'}$ dibentuk dengan menggabungkan θ_v dan $\theta_{v'}$.

Penggabungan kedua model topik dilakukan dengan terlebih dahulu menyamakan dimensi keduanya. Selanjutnya adalah *sampling* C_{dg} dari topik kolaborasi z_{dg} berdasarkan distribusi $\vartheta_{vv'}$ yang baru. Berbeda dengan LDA terhadap *bag-of-words*, perhitungan nilai probabilitas posterior pada CTL Berbasis Frase memiliki batasan bahwa setiap kata pada frase yang sama akan dikaitkan pada topik yang sama. Oleh karena itu, proses pengambilan sampel nilai *s* dan *z* dilakukan sekali untuk setiap frase. Notasi yang digunakan dalam fase ini dirangkum dalam Tabel 3.4.

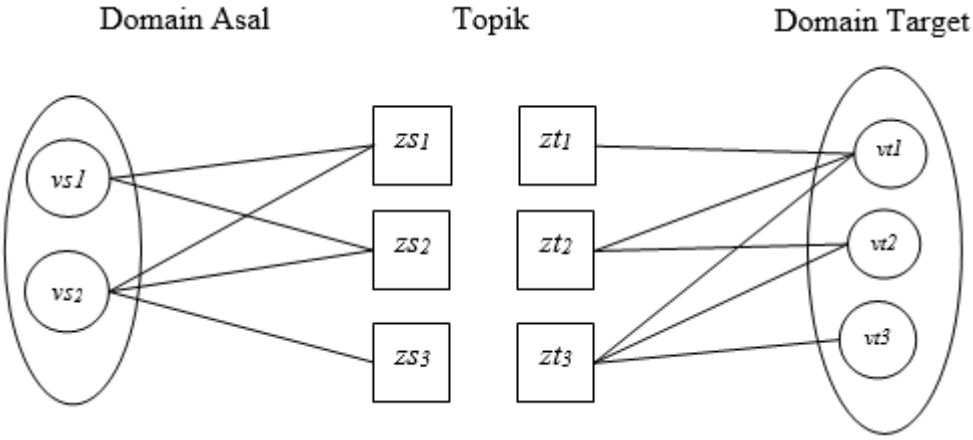
Tabel 3.4 Notasi CTL Berbasis Frase

Simbol	**Deskripsi**
T	Himpunan topik
d	Dokumen kolaborasi
A_d	Himpunan penulis untuk dokumen *d*
X_{dg}	Frase ke *g* pada dokumen *d*
x_{dgj}	Token ke-*j* pada dokumen *d* frase ke *g*
z_{dgj}	Topik yang dikaitkan dengan x_{dgj}
$C_{d,g}$	$\{z_{d,g,j}\}_{j=1}^{\lvert X_{d,g}\rvert}$ yaitu kumpulan topik pada frase ke *g* di dokumen *d*
s_{dgj}	Bernilai 1 jika x_{dgj} adalah kata yang termasuk *single domain* atau dan 0 jika termasuk *cross domain*
θ dan θ'	Distribusi multinomial dari topik pada domain asal dan domain target
θ_v	Distribusi multinomial dari topik spesifik terhadap penulis *v*
$\vartheta_{vv'}$	Distribusi multinomial dari topik spesifik terhadap pasangan penulis (*v, v'*)
ϕ_z	Distribusi multinomial dari kata spesifik terhadap topik *z*
α, β	Parameter Dirichlet
λ	Parameter untuk *sampling* variabel *s*
γ, γ_t	Parameter beta untuk menghasilkan nilai λ

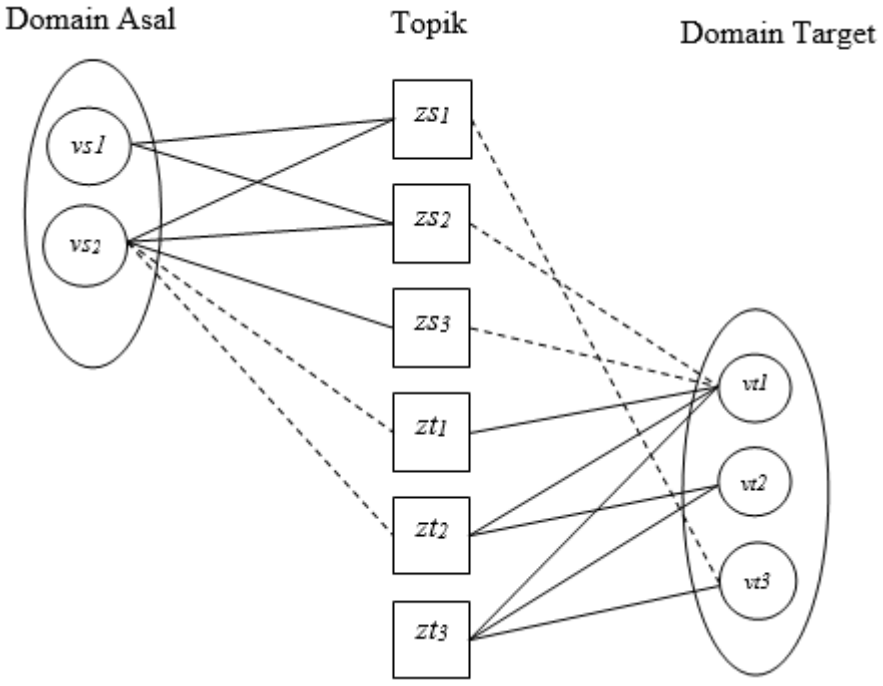


Gambar 3.7 Diagram Fase CTL Berbasis Frase

Gambar 3.8 menunjukkan proses pembentukan model topik untuk domain asal dan domain target dengan *vs* dan *zs* adalah peneliti dan topik dari domain asal, serta *vt* dan *zt* adalah peneliti dan topik dari domain target. Model topik domain asal θ_v, model topik domain target θ'_v, dan data publikasi kolaborasi kemudian dijadikan masukan dalam pembentukan model topik kolaborasi ϑ dari kedua domain yang ditunjukkan pada Gambar 3.9. Proses *cross-domain topic learning* tersebut akan menghasilkan nilai keterkaitan antara peneliti pada domain asal dengan topik penelitian dari domain target dan begitu juga sebaliknya.



Gambar 3.8 Gambaran Hasil Pembentukan Model Topik pada Domain Asal dan Target



Gambar 3.9 Gambaran Hasil *Cross-Domain Topic Learning* Berbasis Frase

Metode ini menggunakan *Gibbs Sampling* untuk mengestimasi nilai parameter $\{\theta, \theta', \vartheta, \phi, \lambda\}$.

1. Probabilitas posterior pada *z* (atau *z'*) untuk setiap kata pada dokumen publikasi oleh penulis dari *single* domain menggunakan formula 3.3. Hasil perhitungan ini akan mempengaruhi nilai θ (atau θ').

$$P\left(C_{dg} = z\middle|x,.\right) = \sum_{j=1}^{X_{dg}}(\alpha + n_{vz_{dgj}}^{-dgj} + j - 1) \times \frac{m_{z_{dgj}x_{dgj}}^{-dgj} + \beta}{\sum_x\left(m_{z_{dgj}x}^{-dgj} + \beta\right) + j - 1} \quad (3.3)$$

n_{vz} : Jumlah berapa kali topik *z* menjadi label untuk penulis *v*

m_{zx} : Jumlah berapa kali kata *x* dilabeli topik *z*

n^{-dgj} : Notasi *–dgj* berarti jumlah tidak memperhitungkan item yang saat ini diproses

2. Probabilitas posterior pada *s* dihitung dengan formula 3.4 dan hasilnya digunakan untuk memperoleh nilai parameter $\theta, \theta', \vartheta$ dengan formula 3.5. Selanjutnya nilai ϕ, λ dapat disimpulkan dari model topik yang dibentuk. Formula 2.6 bisa disesuaikan untuk $P\left(s_{dgj} = 1\middle|.\right)$. Perubahan yang perlu diperhatikan adalah mengganti $\left(n_{vz_{dgj}} + n_{v'z_{dgj}}\right)$ dengan penulis tunggal $n_{vz_{dgj}}$ atau $n_{v'z_{dgj}}$.

$$P\left(s_{dg} = 0\middle|z,.\right) = \sum_{j=1}^{X_{dg}}\left(\alpha + n_{vv'z_{dgj}}^{-dgj} + \left(n_{vz_{dgj}} + n_{v'z_{dgj}}\right) + j - 1\right) \times \frac{n_{ds_0}^{-dgj} + \gamma_t}{n_{ds_0}^{-dgj} + n_{ds_1}^{-dgj} + \gamma_t + \gamma} \quad (3.4)$$

n_{ds_0} : Jumlah berapa kali 0 menjadi sampel pada dokumen *d*

(v, v'): Pasangan penulis yang dipilih untuk suatu x_{dgj}

$n_{vv'z}$: Jumlah berapa kali topik *z* dijadikan label untuk (v, v')

Probabilitas posterior topik *z* didefinisikan pada formula 3.5 sebagai berikut:

$$P\left(C_{dg} = z\middle|s_{dg} = 0, x,.\right) = \sum_{j=1}^{X_{dg}}(\alpha + n_{vv'z_{dgj}}^{-dgj} + (n_{vz_{dgj}} + n_{v'z_{dgj}})) \times \frac{m_{z_{dgj}x_{dgj}}^{-dgj} + m_{z_{dgj}x_{dgj}} + m'_{z_{dgj}x_{dgj}} + \beta}{\sum_x\left(m_{z_{dgj}x}^{-dgj} + m_{z_{dgj}x} + m'_{z_{dgj}x} + \beta\right)} \quad (3.5)$$

m_{zx}^{-dgj}: Jumlah berapa kali kata *x* dilabeli topik *z* pada publikasi kolaborasi

m_{zx} : Jumlah berapa kali kata *x* dilabeli topik *z* di publikasi domain asal

m'_{zx} : Jumlah berapa kali kata *x* dilabeli topik *z* di publikasi domain target

Hasil fase *Cross-Domain Topic Learning* Berbasis Frase adalah model topik untuk setiap peneliti dari domain asal dan domain target.

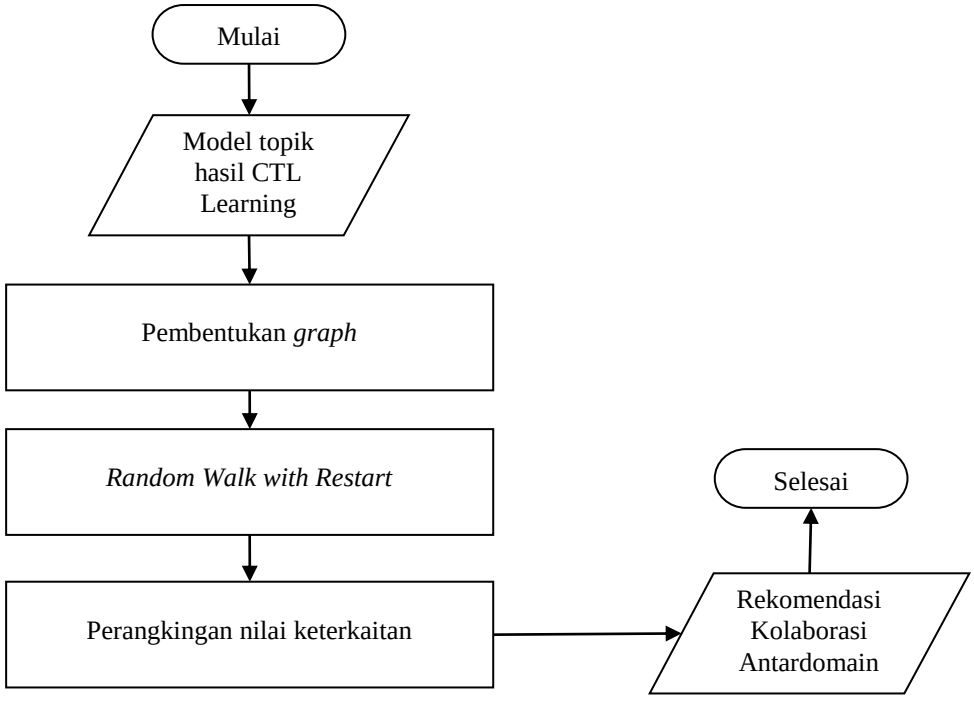
3.3.4. Fase Perankingan Rekomendasi dengan *Random Walk with Restart* (RWR)

Hasil dari fase CTL *Learning* selanjutnya dijadikan dasar dalam perangkingan rekomendasi kolaborasi. Diagram alur fase perangkingan ditunjukkan pada Gambar 3.10. Adapun langkah-langkahnya dijelaskan sebagai berikut:

Langkah 1: Pembentukan *graph* berdasarkan hasil *CTL Learning.* Suatu *node* penulis dihubungkan dengan suatu *node* topik jika memiliki probabilitas posterior $P(z|s = 0, .)$ lebih besar dari *threshold.* Semakin kecil *threshold* yang diberlakukan maka *graph* yang terbentuk akan semakin padat.

Langkah 2: Penghitungan nilai keterkaitan antara penulis-penulis pada domain target dengan *query node* penulis pada domain asal. Nilai keterkaitan diperoleh berdasarkan algoritma *Random Walk with Restart* hingga dicapai konvergensi.

Langkah 3: Perangkingan penulis pada domain target berdasarkan nilai keterkaitan terbesar untuk menentukan rekomendasi kolaborasi yang diberikan.



Gambar 3.10 Diagram Fase Perankingan Rekomendasi

Fase terakhir menghasilkan ranking rekomendasi peneliti pada domain target untuk setiap peneliti pada domain asal. Gambaran hasil rekomendasi ditunjukkan pada Tabel 3.5.

Tabel 3.5 Gambaran Hasil Rekomendasi Kolaborasi

Peneliti Domain Asal	**Rekomendasi Kolaborator Domain Target**
vs_1	vt_1 vt_3
vs_2	vt_1 vt_2 vt_3

3.4. Skenario Uji Coba

Uji coba sistem dilakukan untuk mengukur performa model rekomendasi kolaborasi antardomain terhadap data uji, yaitu judul, abstrak, dan peneliti dari publikasi pada sepasang domain yang dipilih dari dataset *Arnetminer*, yaitu *Visualization* sebagai domain asal dan *Data Mining* sebagai domain target. Domain asal merupakan domain peneliti yang dijadikan sebagai *input query* sedangkan domain target adalah domain dari kolaborator yang hendak dicari.

Proses uji coba dilakukan dengan membagi dataset menjadi dua bagian. Penelitian yang dipublikasikan pada tahun 2000 atau sebelumnya dijadikan data latih (data untuk pembangunan model) dan penelitian yang dipublikasikan setelah tahun 2000 dijadikan data uji (data untuk validasi model). Jika rekomendasi yang diberikan sistem berdasarkan data latih kemudian terlaksana, maka rekomendasi ini dianggap rekomendasi yang benar, dan begitu juga sebaliknya.

Performa model yang diusulkan akan dibandingkan dengan metode rekomendasi dasar *Cross-Domain Topic Learning*. Di samping itu pengujian juga dilakukan untuk mengestimasi nilai *minimum support,* jumlah topik, dan nilai koefisien pada CTL yang menghasilkan sistem rekomendasi kolaborasi antardomain paling optimal.

3.5. Analisa Hasil

Kualitas hasil rekomendasi kolaborasi antardomain dievaluasi berdasarkan parameter *precision* dan *recall*. *Precision* menunjukkan tingkat relevansi rekomendasi yang dihasilkan sedangkan *recall* menujukkan tingkat efektivitas sistem dalam mengidentifikasi keseluruhan rekomendasi. *Precision* dan *recall* dikalkulasi dengan

formula 3.6 dan formula 3.7 dimana $R_{correct}$ adalah jumlah rekomendasi yang teridentifikasi dengan tepat, $R_{incorrect}$ adalah jumlah rekomendasi yang tidak teridentifikasi dengan tepat, sedangkan $R_{missing}$ adalah jumlah rekomendasi yang tidak berhasil diidentifikasi. Semakin tinggi nilai *precision* dan *recall* pada setiap proses maka semakin baik pula model sistem rekomendasi kolaborasi antardomain yang diusulkan.

$$presicion = \frac{R_{correct}}{R_{correct} + R_{incorrect}} \quad (3.6)$$

$$recall = \frac{R_{correct}}{R_{correct} + R_{missing}} \quad (3.7)$$

Performa metode yang diusulkan akan dibandingkan dengan metode dasar CTL berdasarkan beberapa parameter berikut:

1. P@10, yaitu nilai *precision* pada 10 rekomendasi teratas
2. P@20, yaitu nilai *precision* pada 20 rekomendasi teratas
3. R@100, yaitu nilai *recall* dari 100 rekomendasi teratas

3.6. Penulisan Laporan

Pada tahap ini dilakukan pendokumentasian seluruh proses yang dilakukan dalam penelitian, termasuk dasar teori, desain dan implementasi metode usulan, serta hasil dan analisa pengujian. Penyusunan laporan akhir bertujuan untuk memberikan gambaran mengenai pengerjaan penelitian sehingga dapat digunakan sebagai literatur dalam pengembangan metode rekomendasi kolaborasi antardomain berikutnya.

BAB 4
IMPLEMENTASI DAN PENGUJIAN

Pada bab ini dipaparkan implementasi dan uji coba penelitian yang telah dilakukan terhadap metode rekomendasi kolaborasi penelitian antardomain yang diusulkan.

4.1. Spesifikasi Perangkat Pengujian

Metode *Cross-Domain Topic Learning* berbasis frase diimplementasikan menggunakan perangkat keras dan perangkat lunak dengan spesifikasi pada Tabel 4.1. Penelitian ini juga menggunakan modul ToPMine yang sudah dibangun oleh Ahmed El-Kishky dkk dengan bahasa pemrograman Java.

Tabel 4.1 Spesifikasi Perangkat Pengujian

Nama	Spesifikasi
Processor	Intel(R) Core(TM) i5-4200U CPU @1.60GHz 2.30 GHz
RAM	8.00 GB
Sistem Operasi	Microsoft Windows 10 Pro 64-bit
Bahasa Pemrograman	Java (*Preprocessing* dan ekstraksi frase) C# (Proses persiapan data dan *Cross-Domain Topic Learning* Berbasis Frase) Matlab (*Random Walk with Restart*)
Tools	MALLET (*Machine Learning for Language Toolkit*)
Database Management System	MySQL 5.7

4.2. Persiapan Data Uji Coba

Tahap ini dilakukan untuk mempersiapan data *Arnetminer*[1]sesuai dengan kebutuhan sistem. Domain yang dipilih dari dataset *Arnetminer.org* untuk dilakukan pengujian adalah *Visualization* sebagai domain asal dan *Data Mining* sebagai domain target. Data *Arnetminer* terkait kolaborasi tersedia dengan format seperti pada Tabel 4.2. Berikutnya informasi pada kolom *authors* kemudian dipecah menjadi tabel baru yang menyimpan relasi antara publikasi dan peneliti. Format tabel hasil ekstraksi

[1] https://aminer.org/collaboration

ditunjukkan pada Tabel 4.3. Karena tidak adanya informasi *id author*, maka diasumsikan bahwa nama yang berbeda mengindikasikan peneliti yang berbeda. Pada tahap ini juga dilakukan normalisasi nama peneliti yang memiliki aksen khusus seperti à, ñ, dan æ.

Tabel 4.2 Format Data Awal *Arnetminer*

id	domain	journal/ conf	title	authors	year	abstract
1	Data Mining	KDD	Discriminant Adaptive Nearest Neighbor Classification.	Trevor Hastie,Robert Tibshirani	1995	Nearest neighbor classification expects the class conditional probabilities ...
2	Data Mining	KDD	A Perspective on Databases and Data Mining.	Marcel Holsheimer,Martin L. Kersten,Heikki Mannila,Hannu Toivonen	1995	We discuss the use of database methods for data mining. Recently impressive ...
3	Data Mining	KDD	New Algorithms for Fast Discovery of Association Rules.	Mohammed Javeed Zaki,Srinivasan Parthasarathy,Mitsunori Ogihara,Wei Li	1997	Association rule discovery has emerged as an important problem in knowledge ...
4	Visualization	IEEE Vis	Dynamic Graphics for Network Visualization.	Richard A. Becker,Stephen G. Eick	1999	Network data involves statistics that are associated with the nodes or links in a network ...
5	Visualization	IEEE Vis	Shape Coding of Multidimensional Data on a Microcomputer Display.	J. Beddow	1990	The visual representation of data from complex systems, whether databases, measured scientific data, or simulation output, …

Tabel 4.3 Tabel Relasi *Paper* dan *Authors*

paperid	authorName	authorNamePlain
1	Trevor Hastie	trevor hastie
1	Robert Tibshirani	robert tibshirani
2	Marcel Holsheimer	marcel holsheimer
2	Martin L. Kersten	martin l. kersten
2	Heikki Mannila	heikki mannila
2	Hannu Toivonen	hannu toivonen
4	Richard A. Becker	richard a becker
4	Stephen G. Eick	stephen g eick
5	J. Beddow	j beddow

Selanjutnya data dibagi menjadi data latih dan data uji berdasarkan tahun publikasi. Penelitian yang dipublikasikan pada tahun 2000 dan sebelumnya dijadikan

sebagai data latih dan penelitian yang dipublikasikan setelah tahun 2000 dijadikan data uji. Proses pengolahan ini menghasilkan input data dalam format file teks sebagai berikut:

1. **Papers.txt** menyimpan judul dan abstrak seluruh publikasi, yaitu publikasi domain asal, publikasi domain target, dan publikasi kolaborasi.
2. **Authors.txt** menyimpan ID peneliti dari publikasi baik peneliti domain asal maupun peneliti domain target dengan urutan yang sama dengan Papers.txt.
3. **SourceAuthorIDs.txt** menyimpan daftar ID dan nama peneliti domain asal.
4. **TargetAuthorIDs.txt** menyimpan daftar ID dan nama peneliti domain target.
5. **Info.txt** menyimpan detil informasi jumlah penulis domain asal dan domain target serta jumlah publikasi domain asal, domain target, dan publikasi kolaborasi.
6. **SourceAuthorTested.txt** menyimpan daftar ID peneliti dari domain asal yang hendak dicari kolaboratornya. Format file sourceAuthorTested.txt ditunjukkan pada Gambar 4.1.

```
85
158
495
738
1084
1322
1356
1357
1363
1387
1427
....
....
```

Gambar 4.1 Contoh Format File SourceAuthorTested.txt

7. **GroundTruth.txt** menyimpan daftar kolaborasi penelitian pada data uji yang selanjutnya digunakan sebagai *ground truth* pada tahap uji coba. Format file GroundTruth.txt ditunjukkan pada Gambar 4.2. Pada baris pertama misalnya, untuk peneliti pada domain asal dengan id 85 (baris pertama Gambar 4.1), kolaborasi pada data uji dilakukan dengan peneliti dari domain target dengan id 2010, 2011, 2012, 2443, 2444, dan 2695 (baris pertama Gambar 4.2). Hal yang sama berlaku untuk baris-baris lain pada kedua file dengan nomor baris yang bersesuaian.

```
2010,2011,2012,2443,2444,2695
1142,1143,1144,1349,1887,1888,1889,1890
287,384,496,917,918,1840,1841,1891,1892,1893
38,39,40,961,1481,1655,1822,1823,2205,2206,2207,2325,2326,2333,2334,2647,2734,2762,2
785,2825,2826,2830,2831
1520,1521,1522,1523,1928
74,1819,1820,2104,2105,2106,2528
908,909,946,1362,1363,1962
720,908,909,1362,1363,1724,1725,1957
```

Gambar 4.2 Contoh Format File GroundTruth.txt

Tabel 4.4 menunjukkan beberapa informasi mengenai hasil pengolahan data baik dari domain asal maupun domain target.

Tabel 4.4 Informasi Data Set

Keterangan	**Jumlah**
Jumlah peneliti domain asal	5399
Jumlah peneliti domain target	3274
Jumlah publikasi domain asal	3862
Jumlah publikasi domain asal pada data latih	2390
Jumlah publikasi domain target	2190
Jumlah publikasi domain target pada data latih	1175
Jumlah publikasi kolaborasi	535
Jumlah publikasi kolaborasi data latih	210
Jumlah publikasi kolaborasi data uji	325
Jumlah peneliti domain asal yang memiliki kolaborasi penelitian antardomain pada data latih	256
Jumlah peneliti domain asal yang memiliki minimal 3 kolaborasi penelitian antardomain pada data uji	99
Jumlah peneliti domain asal yang memiliki kolaborasi penelitian antardomain pada data latih dan minimal 3 kolaborasi penelitian pada data uji	57

4.3. Implementasi Proses *Training*

Implementasi proses training dilakukan dengan mengembangkan sistem rekomendasi kolaborasi penelitian antardomain yang diusulkan. Sistem yang dibangun terdiri dari empat tahapan utama, yaitu *preprocessing*, ektraksi frase, *Cross-Domain Topic Learning* Berbasis Frase, dan perankingan rekomendasi.

4.3.1. Implementasi *Preprocessing*

Pada fase ini dilakukan tahapan *text preprocessing* terhadap seluruh data publikasi pada file Papers.txt dengan modul ToPMine yang telah tersedia. Tahap *preprocessing* melingkupi *tokenizing*, *stopword and rare word removal*, dan *stemming*. Hasil akhir dari tahap ini adalah sebagai berikut:

1. **Input_vocFile.txt** merupakan kamus kata dengan format seperti pada Gambar 4.3. Kolom pertama merupakan kata dalam bentuk dasarnya dan kolom kedua merupakan id kata dasar dalam kamus.
2. **Input_stemMapping.txt** menyimpan daftar kata dasar beserta bentuk asli dan frekuensi kemunculannya dengan format seperti pada Gambar 4.4. Pada baris ketiga misalnya, kata dasar *direct* muncul pada korpus dalam bentuk imbuhan *directed* sebanyak satu kali dan *directions* sebanyak dua kali.
3. **Input_phraseFile.txt** yang menyimpan representasi publikasi dalam id kata pada kamus dengan format seperti pada Gambar 4.5. Baris pertama pada file ini adalah informasi jumlah kata dalam kamus dan jumlah dokumen pada korpus. Pada baris selanjutnya, setiap baris merupakan satu dokumen. Angka-angka yang dipisahkan tanda baca koma merupakan id dari kata dasar setiap token.

```
data     0
base     1
direct   2
inform   3
resource 4
comput   5
manag    6
```

Gambar 4.3 Contoh Format Input_vocFile.txt

```
data    :  data 14
base    :  based 1     base 3
direct  :  directed 1     directions 2
inform  :  information 6
resourc :  resource 3
comput  :  computer 3    computing 1  computational 1
manag   :  management 4
```

Gambar 4.4 Contoh Format Input_stemMapping.txt

```
vocabSize:38      docNum:7
0,1,2,3,4,5,5,6,0,5,7,0,6,0,1,2,3,4,6
0,8,1,9,10,5,8,11,0,8,11
12,13,14,15,16,7,0,9,17,5,16,14,15,18,12,13,15,16,7,19,17,15,16,0,18
,15,20,12,13,15,16,7,12,17,10,21,22,22
23,24,25,26,27,20,23,24,25,26,23,20,18,27,25,26,27,20,24,25,26,27,26
29,30,31,29,30,0,1,29,30,14,27,31,10,9,20,32,29,30,22,22,31
33,3,17,34,7,33,3,0,7,19,34,7,6,19,2,34,8,20,33,3,34,8,33,3,0,28,9,1
35,36,7,10,21,35,16,7,10,21,37,0,28,37,36,10,20,32,28,36,8,35,7,10,2
1,0,28,9,28,7,37,28,32,35,0,28,34,7,32,30,36,35,9,28
```

Gambar 4.5 Contoh Format Input_phraseFile.txt

4.3.2. Implementasi Ekstraksi Frase

Tahap ini bertujuan untuk mentransformasi publikasi dari format *bag-of-words* menjadi format *bag-of-phrases.* Proses ekstraksi frase terdiri dari dua tahapan utama, yaitu *frequent phrase mining* dan *phrase construction*. *Frequent phrase mining* dilakukan dengan menghitung jumlah kemunculan frase dengan panjang 1 hingga *n* yang memenuhi *minimum support.* Potongan kode program untuk tahapan ini ditunjukkan pada Gambar 4.6.

```
public int mineFixedPattern(int lastDocument, int patternSize)
{
      int index = 0;
      Counter<Counter<Integer>> insufficientPatterns = new
        Counter<Counter<Integer>>();

      while (index <= lastDocument){
            int[] doc = documents[index];
            BitSet docApriori = apriori[index];
            boolean continueMining = this.mine(doc, patternSize,
               insufficientPatterns, docApriori);

            if (!continueMining || doc.length <= patternSize){
                  documents[index] =  documents[lastDocument];
                  apriori[index] = apriori[lastDocument];
                  lastDocument -= 1;
            }
            Else{
                  index +=1;
            }
      }

      insufficientPatterns.clear();
      insufficientPatterns = null;
      patternSize += 1;
      System.out.println("Documents remaining : "+lastDocument);
      System.gc();
      return lastDocument;
}
```

Gambar 4.6 Potongan Kode Program *Frequent Phrase Mining*

Selanjutnya dilakukan proses *phrase construction* untuk mengeliminasi frase yang secara kebetulan diidentifikasi sebagai *frequent phrase* meskipun bukan merupakan frase yang valid. Frase dibentuk berdasarkan nilai *significance score* terhadap pasangan kata yang berurutan. Potongan kode program *phrase construction* ditunjukkan pada Gambar 4.7.

```
public double significance(Counter<Integer> pattern)
{
     int actualOccurence = this.patterns.get(pattern);
     double independentProb = 1;
     ArrayList<Integer> phrase = pattern.getAll();

     for (int word : phrase)
     {
         Counter<Integer> wordInstance = new Counter<Integer>();
         wordInstance.add(word);
          independentProb *= ((double)this.patterns.get(wordInstance))
            /this.numWords;
     }

     int factorialIndex = Math.min(this.maxPhrase, phrase.size());
     independentProb *= this.fact[factorialIndex];
     double expectedOccurence = independentProb * (this.numWords -
        phrase.size());
     double variance = expectedOccurence*(1-independentProb);
     double sig = (actualOccurence - expectedOccurence)
        /Math.sqrt(variance);
     System.out.println(expectedOccurence+","+actualOccurence);
     return sig;
}
```

Gambar 4.7 Potongan Kode Program *Phrase Construction*

```
vocabSize:2577      docNum:1002
0 1 ,2 ,3 ,4 ,5 ,6 ,7 ,8 ,9 ,10 11 ,12 ,13 ,14 ,15 ,16 ,17 ,10 ,18 ,19 ,20 ,21
0 ,22 ,23 ,24 ,10 ,25 ,26 ,27 ,28 ,29 ,30 ,0 21 ,31 ,20 ,32 ,33 ,0 1 ,2 ,34
,35 ,34 ,36 ,37 ,38 ,3 ,4 ,21 ,39 ,40 ,41 ,42 ,43

44 ,45 ,46 ,0 47 ,1 ,48 49 ,50 ,51 52 ,10 ,47 ,53 ,54 ,0 47 ,55 ,56 ,57 ,58
,59 ,54 ,60 ,61

62 ,63 ,64 ,65 ,66 ,67 ,68 ,69 ,28 ,0 ,48 ,29 ,70 ,10 ,71 ,69 ,72 ,73 ,66 ,74
,75 ,76 ,77 ,78 ,79 ,80 ,67 ,68 ,81 ,82 ,83 ,84 ,85 ,86 ,87 ,88 ,89 ,82 ,90
,91 ,92 ,64 ,65 ,67 ,69 ,28 93 ,94 ,95 ,96 ,70 ,97 ,67 ,69 ,0 ,98 ,91 ,99 ,100
,101 ,102 ,103 ,31 ,104 ,67 ,105 ,106 ,107 ,108 109 ,62 ,63 ,64 ,65 ,67 ,69
,28 ,110 ,64 ,111 ,112 ,70 ,113 ,114 ,115 ,116 ,117 ,52 118 ,119 ,120 ,121
,122 ,123 ,124 ,119 ,125 ,126 ,127

128 ,129 ,130 ,131 ,132 ,108 ,128 ,133 ,5 ,127 ,134 ,129 ,130 ,131 ,128 ,108
,135 ,136 ,91 ,77 ,137 ,19 ,132 ,138 ,139 ,140 ,141 ,130 ,131 ,142 ,132 ,108
,143 ,129 ,130 ,131 ,132 ,131 ,4 ,130 ,144 ,145 ,130 ,117 ,142 ,146 ,147 ,148

149 150 ,151 ,152 ,149 150 ,45 ,153 ,154 ,0 1 ,155 ,156 ,157 ,158 ,149 150 ,76
,159 ,160 ,161 ,162 ,163 ,66 ,164 ,132 ,155 ,152 ,165 ,52 ,48 ,108 ,166 ,94
,149 150 ,119 ,119 ,152 ,167 ,168 ,169 ,136 ,170 ,171 ,172
```

Gambar 4.8 Contoh Hasil *Phrase Extraction*

Hasil akhir dari tahap ini adalah *bag-of-words* dari publikasi. Contoh hasil fase *phrase extraction* dalam format *bag-of-words* yang diperoleh ditunjukkan pada Gambar 4.8. Dapat dilihat bahwa jika beberapa rangkaian kata termasuk frase, maka tanda baca koma yang memisahkannya akan dihapuskan seperti pada angka-angka yang dicetak

tebal. Keseluruhan data ini selanjutnya dijadikan input dalam proses *Cross-Domain Topic Learning* Berbasis Frase.

4.3.3. Implementasi *Cross-Domain Topic Learning* Berbasis Frase

Pada fase ini dilakukan pemodelan topik terhadap peneliti pada domain asal dan publikasi domain target. Pada masing-masing domain, hasil dari proses ini adalah distribusi probabilitas keterkaitan peneliti terhadap topik dan distribusi probabilitas keterkaitan kata terhadap topik. Kedua variabel tersebut disimpan dalam file json dengan format seperti pada Gambar 4.9. Setiap *bracket* merupakan distribusi probabilitas keterkaitan seorang peneliti terhadap seluruh topik. Jika terdapat 30 topik misalnya, maka akan terdapat 30 baris nilai probabilitas untuk masing-masing peneliti. Angka yang dicetak tebal misalnya, menunjukkan bahwa nilai probabilitas keterkaitan peneliti pertama (*bracket* pertama) terhadap topik keempat (baris keempat) adalah 0.0395.

```
"theta": [
  [
   0.00048780487804878054,
   0.00048780487804878054,
   0.00048780487804878054,
   0.03951219512195122,
   0.00048780487804878054,
            ...
            ...
  ],
  [
   0.00019342359767891685,
   0.044680851063829789,
   0.0021276595744680851,
   0.00019342359767891685,
   0.029206963249516441,
            ...
            ...
  ],
  [...]
]
```

Gambar 4.9 Format Distribusi Probabilitas Peneliti Terhadap Topik

Selanjutnya, kedua model topik yang dihasilkan dari domain asal dan domain target kemudian dijadikan input dalam ekstraksi model topik pada publikasi kolaborasi menggunakan *Cross-Domain Topik Learning* Berbasis Frase. Potongan kode program pada CTL Berbasis Frase ditunjukkan pada Gambar 4.10.

```
private int DoSampling()
{
  ... ... ...

  for (int docID = 0; docID < M; docID++)
  {
     var currDoc = cor.Docs[docID];
     var groups = currDoc.WordGroups;

     for (int g = 0; g < groups.Length; g++)
     {
        var group = groups[g];
        ... ... ...

        for (int k = 0; k < K; k++)
        {
            p[k] = (zvv[vs, vt, k] + zvSource[vs][k]
                           + zvTarget[vt][k] + alpha);

            for (int wID = 0; wID < group.Length; wID++)
            {
               int w = group[wID];
               p[k] *= (zx[w][k] + zxSource[w][k] + zxTarget[w][k]
                             + beta) / (zxsum[k] + zxsumSource[k]
                              + zxsumTarget[k] + Vocbeta);
            }

        ... ... ...
        }
    }
}
```

Gambar 4.10 Potongan Kode Program *Gibbs Sampling* ACT

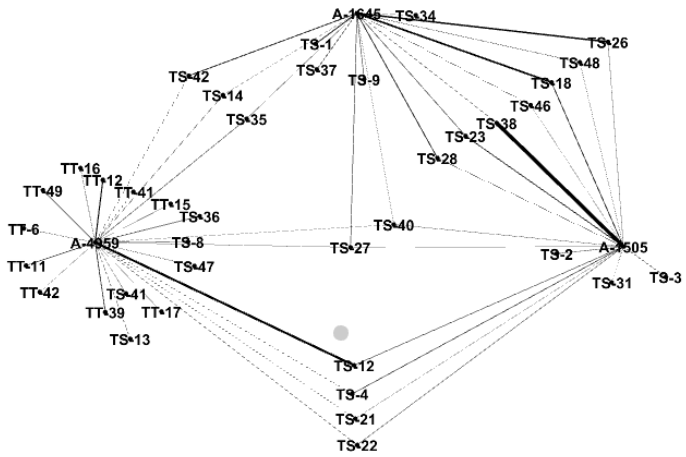
Proses ini memungkinkan peneliti pada domain target memiliki distribusi probabilitas terhadap topik pada domain target dan juga sebaliknya. Nilai tersebut kemudian disimpan dalam file dataForMatlab.txt dengan format seperti pada Gambar 4.11. Angka pada kolom pertama menunjukkan id peneliti, angka pada kolom kedua menunjukkan id topik, dan angka ketiga merupakan nilai probabilitas keterkaitan peneliti dan topik. Daftar nilai probabilitas tersebut selanjutnya dijadikan sebagai input pada proses perangkingan dengan *Random Walk with Restart.*

```
1    8677    0.0267543859649123
1    8680    0.0223684210526316
1    8682    0.0267543859649123
1    8684    0.0618421052631579
1    8685    0.0135964912280702
1    8692    0.0135964912280702
1    8693    0.0135964912280702
1    8694    0.0706140350877193
....
....
```

Gambar 4.11 Format Data Input Perangkingan

4.3.4. Implementasi Perangkingan Rekomendasi

Topik model yang dihasilkan dari tahap *cross-domain topic learning* selanjutnya digunakan untuk membentuk graf dengan setiap penulis dan topik sebagai *node* dan nilai pobabilitas sebagai *weight.* Graf yang terbentuk dapat divisualisasikan seperti pada Gambar 4.12. Contoh ini difokuskan pada tiga peneliti acak yang saling memiliki keterkaitan melalui topik-topik, baik topik pada domain asal maupun topik pada domain target. Node dengan awalan A (*Author*) menunjukkan node peneliti, awalan TS (*Topic Source*) menunjukkan topik pada domain asal, dan awalan TT (*Topic Target*) menunjukkan topik pada domain target. Setiap *edge* yang menghubungkan *node* peneliti dan topik memiliki bobot sebesar nilai probabilitas keterkaitan peneliti dengan topik tersebut.



Gambar 4.12 Graf Keterkaitan Peneliti dan Topik

Rekomendasi untuk setiap peneliti pada domain asal kemudian ditentukan dengan metode *Random Walk with Restart.* RWR hanya dijalankan dengan *query* peneliti-peneliti pada domain asal yang memiliki kolaborator pada domain asal pada data uji. Kode program perankingan rekomendasi ditunjukkan pada Gambar 4.13. Hasil dari proses ini adalah urutan kolaborator pada target domain yang memiliki nilai keterkaitan tertinggi seperti pada Gambar 4.14. Setiap baris merupakan hasil rekomendasi untuk setiap peneliti pada domain asal yang dijadikan *query.* Baris pertama misalnya, menunjukkan hasil rekomendasi kolaborator pada domain target untuk peneliti pertama yang diuji dari domain asal. Urutan rekomendasi yang diperoleh adalah peneliti domain target dengan id 2710, 3223, 2011, 1595, dan seterusnya.

```
function [pi,time,numiter]=rwr(pi0,H,v,n,alpha,epsilon)

tic;
S=load('data.txt');
n = max(max(S));
M=sparse(S(:,1),S(:,2),S(:,3), n, n);
M=full(M);
H = M' + M;
H(1:n+1:end)=diag(M);
epsilon = 1e-4;
n = size(H,1);
H = normr(H);
v = zeros(1,n);
v(1,1) = 1;
alpha = 0.5;

k=0;
residual=1;
pi=pi0;

while (residual > epsilon)
  prevpi=pi;
  k=k+1;
  pi=(1-alpha)*pi*H + alpha*v;
  residual=norm(pi-prevpi,1);
end

numiter = k;
toc;
```

Gambar 4.13 Kode Program *Random Wak with Restart*

```
2710,3223,2011,1595,353,146,1098,1523,873,1705,1887,2525,2036,...
1887,3180,872,956,2516,738,218,2519,3088,2031,2518,1098,2112,...
93,204,495,1098,915,1893,1719,152,146,496,2710,2289,652,3082,...
146,495,1103,2206,2647,1102,1550,40,1549,2334,1840,496,39,1790,...
1523,1705,1887,2878,146,1098,873,2289,1595,2966,652,353,2580,...
2546,3180,1820,1149,1275,1180,2847,1191,1819,507,1603,1804,2604,...
```

Gambar 4.14 Hasil Perangkingan dengan *Random Wak with Restart*

4.4. Pengujian

Dalam proses pengujian, dilakukan perbandingan antara hasil rekomendasi dari algoritma *Random Walk with Restart* pada Gambar 4.14 dengan GroundTruth.txt (Gambar 4.2) untuk setiap peneliti pada domain asal yang diuji (file sourceAuthorTested.txt pada Gambar 4.1). Ketiga file tersebut berkaitan sesuai dengan nomor baris yang sama pada setiap file. Misalnya pada baris pertama, untuk peneliti domain asal dengan ID 85, kolaborasi pada data uji dilakukan dengan peneliti dari domain target dengan id 2010, 2011, 2012, 2443, 2444, dan 2695. Sementara rekomendasi yang diberikan sistem adalah 2710, 3223, 2011, 1595, dan seterusnya.

Rekomendasi yang benar merupakan irisan dari *ground truth* dan rekomendasi yang dihasilkan sistem. Untuk peneliti pertama, rekomendasi kolaborator yang benar hanya satu, yaitu peneliti dari domain target dengan id 1887.

Pada 10 rekomendasi teratas, nilai *precision* untuk peneliti domain asal dengan id 85 diperoleh dengan perhitungan berikut:

$$presicion = \frac{R_{correct}}{R_{recommendation}} = \frac{1}{10} = 0.1$$

Dengan jumlah *ground truth* untuk peneliti domain asal dengan id 85 berjumlah 6, nilai *recall* diperoleh dengan perhitungan berikut:

$$recall = \frac{R_{correct}}{R_{groundTruth}} = \frac{1}{6} = 0.17$$

Keseluruhan nilai *precision* dan *recall* merupakan rata-rata dari setiap *precision* dan *recall* dari seluruh peneliti domain asal yang diuji.

4.4.1. Pengujian Nilai *Minimum Support*

Uji coba ini dilakukan untuk mengetahui pengaruh nilai *minimum support* pada proses ekstraksi frase terhadap hasil rekomendasi. Uji coba dilakukan beberapa kali pada nilai *minimum support* 30, 40, 50, dan 60. Hasil uji coba ditunjukkan pada Tabel 4.5. Nilai *precision* dan *recall* tidak secara seragam berubah sesuai dengan perubahan nilai *minimum support.* Nilai *precision* dan *recall* terbaik dihasilkan dengan *minimum support* 50 sedangkan nilai *precision* dan *recall* terendah dihasilkan dengan *minimum support* 60.

Tabel 4.5 Uji Coba Nilai *Minimum Support*

MinSup	Top 10		Top 20		Top 100	
	Precision Terbaik (%)	P@10 (%)	Precision Terbaik (%)	P@20 (%)	Recall Terbaik (%)	R@100 (%)
30	60.00	9.12	40.00	6.49	100.00	37.38
40	50.00	7.72	30.00	6.23	100.00	36.42
50	**60.00**	**9.47**	**40.00**	**6.67**	**100.00**	**41.53**
60	30.00	7.72	40.00	5.96	100.00	33.32

4.4.2. Pengujian Jumlah Topik

Uji coba ini bertujuan untuk mengetahui jumlah topik yang menghasilkan rekomendasi terbaik. Uji coba dilakukan dengan jumlah topik 30, 35, 40, 45, dan 50 topik dengan beberapa variasi nilai *minimum support.* Hasil uji coba ditunjukkan pada Tabel 4.6. Sama halnya dengan pengujian pada *minimum support*, hasil pengujian terhadap jumlah topik juga tidak spesifik mengikuti tren tertentu, namun *precision* dan *recall* yang terbaik umumnya dicapai pada jumlah topik yang lebih besar.

Tabel 4.6 Uji Coba Jumlah Topik

Parameter		**Top 10**		**Top 20**		**Top 100**	
Minsup	**Topik**	**Precision Terbaik (%)**	**P@10 (%)**	**Precision Terbaik (%)**	**P@20 (%)**	**Recall Terbaik (%)**	**R@100 (%)**
30	30	40.00	6.84	35.00	**6.67**	100.00	30.10
	35	50.00	8.77	50.00	5.96	100.00	31.19
	40	60.00	**9.47**	40.00	6.40	100.00	34.14
	45	**70.00**	9.12	45.00	6.23	100.00	**34.92**
	50	60.00	9.12	40.00	6.49	100.00	37.38
40	30	50.00	6.49	40.00	5.79	100.00	33.08
	35	50.00	8.42	35.00	5.44	100.00	34.11
	40	40.00	8.77	**45.00**	5.61	100.00	35.30
	45	**60.00**	**9.30**	40.00	6.05	100.00	33.97
	50	50.00	7.72	30.00	**6.23**	100.00	**36.42**
50	30	50.00	7.19	40.00	5.00	100.00	29.51
	35	**70.00**	9.30	50.00	5.79	100.00	33.50
	40	40.00	8.25	**45.00**	5.61	100.00	35.10
	45	30.00	9.12	35.00	6.23	100.00	35.19
	50	60.00	**9.47**	40.00	**6.67**	100.00	**41.53**

4.4.3. Pengujian Tahun Pembatas Data Latih dan Data Uji

Berdasarkan pada metode dasar yang dikembangkan yaitu *Cross-Domain Topic Learning*, maka pembagian data latih dan data uji dari ArnetMiner mengikuti penelitian sebelumnya. Pengujian ini dilakukan untuk mengetahui pengaruh penentuan batas tahun dalam pembagian data terhadap performa metode rekomendasi yang diusulkan. Tabel 4.7 menunjukkan jumlah publikasi kolaborasi antardomain pada data latih dan data uji berdasarkan pemilihan tahun batas.

Tabel 4.7 Jumlah Data Uji dan Data Latih

Keterangan	**1997**	**1998**	**1999**	**2000**	**2001**	**2002**	**2003**
Jumlah publikasi kolaborasi data latih	93	126	166	210	253	300	374
Jumlah publikasi kolaborasi data uji	442	409	369	325	282	235	161

Tabel 4.8 Hasil Uji Coba Tahun Pembagian Data Latih dan Data Uji

Tahun Batas	**Top 10**		**Top 20**		**Top 100**	
	Precision Terbaik (%)	**P@10 (%)**	**Precision Terbaik (%)**	**P@20 (%)**	**Recall Terbaik (%)**	**R@100 (%)**
1997	30.00	9.57	35.00	7.61	66.67	27.66
1998	**60.00**	10.43	**55.00**	**9.35**	66.67	30.04
1999	30.00	10.77	25.00	6.73	83.33	28.35
2000	50.00	**12.50**	50.00	8.39	77.78	30.05
2001	40.00	10.00	25.00	6.61	**100.00**	32.82
2002	40.00	10.03	25.00	6.61	**100.00**	**34.00**
2003	40.00	8.40	20.00	6.00	66.67	28.27

Tabel 4.8 menunjukkan hasil uji coba terhadap batas tahun dalam pembagian data latih dan data uji. Pada 100 rekomendasi teratas, nilai recall tidak menunjukkan tren tertentu terhadap perubahan batas tahun. Pada 10 rekomendasi teratas, nilai *precision* terbaik dihasilkan dengan batas tahun 2001 yaitu sebesar 12.5%. Nilai *precision* yang cukup baik juga dihasilkan dengan batas tahun 1998, 1999, 2001, dan 2002. Sementara itu, nilai *precision* terendah didapat pada tahun 1997 dan 2003 dimana perbedaan jumlah data latih dan data uji terlalu besar. Nilai *precision* yang paling

rendah pada 20 rekomendasi juga didapati pada tahun 1997 dan 2003, namun nilai tertinggi dihasilkan pada tahun batas 1998.

4.4.4. Pengujian Koefisien *Alpha, Beta,* dan *GammaT*

Uji coba selanjutnya dilakukan untuk mengetahui pengaruh nilai *alpha, beta,* dan *gammaT* terhadap performa sistem rekomendasi kolaborasi antardomain yang diusulkan. Ketiganya adalah koefisien pada proses pembentukan topik model. Semakin besar nilai *alpha* berarti diasumsikan dokumen dalam korpus terdiri dari banyak topik. Nilai *beta* yang lebih besar berarti diasumsikan setiap topik dibentuk oleh semakin banyak kata. Sementara nilai gammaT merupakan bias yang berpengaruh pada pembentukan topik model pada dokumen kolaborasi.

Pada metode dasar *Cross-Domain Topic Learning,* pengujian dilakukan dengan nilai *alpha* 0.1, *beta* 0.01, dan *gammaT* 3.0 dan disimpulkan bahwa ketiga koefisien tidak berpengaruh besar terhadap performa model. Uji coba ini dilakukan untuk memastikan bahwa model yang dikembangkan juga tidak sensitif terhadap perubahan parameter.

Tabel 4.9 Hasil Uji Coba Koefisien *Alpha*

Alpha	**Top 10**		**Top 20**		**Top 100**	
	Precision Terbaik (%)	**P@10 (%)**	**Precision Terbaik (%)**	**P@20 (%)**	**Recall Terbaik (%)**	**R@100 (%)**
0.1	60.00	9.47	45.00	6.67	100.00	41.53
0.2	60.00	9.82	45.00	5.88	100.00	37.08
0.3	60.00	9.65	40.00	5.96	100.00	36.31
0.4	**50.00**	**8.07**	**40.00**	**5.53**	**100.00**	**31.25**

Tabel 4.10 Hasil Uji Coba Koefisien *Beta*

Beta	**Top 10**		**Top 20**		**Top 100**	
	Precision Terbaik (%)	**P@10 (%)**	**Precision Terbaik (%)**	**P@20 (%)**	**Recall Terbaik (%)**	**R@100 (%)**
0.01	60.00	9.47	45.00	6.67	100.00	41.53
0.02	60.00	9.82	45.00	5.88	100.00	37.08
0.03	60.00	9.65	40.00	5.96	100.00	36.31
0.04	**40.00**	**7.89**	40.00	5.96	100.00	38.17

Tabel 4.11 Hasil Uji Coba Koefisien *GammaT*

GammaT	Top 10		Top 20		Top 100	
	Precision Terbaik (%)	**P@10 (%)**	**Precision Terbaik (%)**	**P@20 (%)**	**Recall Terbaik (%)**	**R@100 (%)**
2.0	50.00	8.07	40.00	5.44	100.00	33.11
3.0	60.00	**9.82**	40.00	5.44	100.00	35.17
4.0	50.00	9.65	35.00	6.40	100.00	37.08
5.0	50.00	8.95	45.00	5.88	100.00	34.65

Berdasarkan hasil uji coba pada nilai *alpha*, *beta*, dan *gammaT*, secara umum metode *Cross-Domain Topic Learning* Berbasis Frase juga tidak sensitif terhadap perubahan nilai ketiga koefisien tersebut. Secara wajar, nilai *precision* mengalami penurunan ketika nilai *alpha* dan *beta* ditentukan terlalu jauh dari nilai yang dipakai pada penelitian sebelumnya, misalnya nilai 0.4 untuk *alpha* yang awalnya 0.1 dan nilai 0.04 untuk *beta* yang awalnya 0.01. Sementara itu, penurunan nilai *gammaT* dari nilai awal 3.0 menjadi 2.0 mengakibatkan penurunan nilai *precision* yang signifikan pada 10 rekomendasi teratas.

4.4.5. Perbandingan dengan CTL

Uji coba juga dilakukan untuk membandingkan performa CTL dengan CTL Berbasis Frase. Hasil uji coba pada Tabel 4.12 menunjukkan bahwa secara umum CTL Berbasis Frase menghasilkan nilai *precision* dan *recall* yang relatif lebih baik dibandingkan dengan CTL. Kinerja CTL lebih baik pada beberapa kasus seperti pada nilai *precision* pada 10 dan 20 rekomendasi terbaik dengan jumlah topik 50.

Tabel 4.12 Perbandingan CTL dan CTL Berbasis Frase

Jumlah Topik	Top 10				Top 20				Top 100			
	Precision Terbaik (%)		P@10 (%)		Precision Terbaik (%)		P@20 (%)		Recall Terbaik (%)		R@100 (%)	
	CTL	CTLBF	CTL	CTLBF	CTL	CTLBF	CTL	CTLBF	CTL	CTLBF	CTL	CTLBF
30	20.00	**50.00**	6.49	**7.19**	30.00	**40.00**	**5.09**	5.00	100	100	**32.95**	29.51
35	30.00	**70.00**	6.84	**9.30**	30.00	**50.00**	4.91	**5.79**	100	100	29.62	**33.50**
40	40.00	**60.00**	8.42	**9.47**	**40.00**	**40.00**	5.44	**6.40**	100	100	**35.96**	34.14
45	60.00	**60.00**	8.77	**9.30**	**45.00**	40.00	5.88	**6.05**	100	100	32.69	**33.97**
50	50.00	**60.00**	**10.35**	9.47	**45.00**	40.00	**7.11**	6.67	100	100	37.40	**41.53**
Rata-rata	40.00	**60.00**	8.17	**8.95**	38.00	**42.00**	5.69	**5.98**	100	100	33.72	**34.53**

4.5. Analisa Hasil

Hasil uji coba menunjukkan bahwa rata-rata *precision* dan *recall* pada rekomendasi yang diberikan sangat rendah. Hal ini salah satunya disebabkan karena tidak tersedianya cukup publikasi untuk peneliti pada data latih sehingga pembentukan distribusi probabilitas topik untuk peneliti tersebut kurang representatif. Tabel 4.13 menunjukkan contoh daftar peneliti yang memiliki nilai *precision* 0% pada CTL berbasis frase dengan jumlah topik 50.

Tabel 4.13 Peneliti dengan Nilai *Precision* 0%

No	Nama	Jumlah Kolaborasi pada Data Uji	Jumlah Publikasi pada Data Latih
1.	Paul R. Cohen	3	1
2.	Steve Lawrence	6	1
3.	David Hart	4	1
4.	Bernhard Scholkopf	5	1

Perbedaan hasil rekomendasi CTL Berbasis Frase dan CTL salah satunya ditunjukkan pada perbedaan nilai *precision* pada peneliti "Vincent Y. Lum". Rekomendasi dengan CTL menghasilkan *precision* 0% sedangkan dengan CTL Berbasis Frase memperoleh *precision* 40%. Gambar 4.15 menunjukkan judul beserta abstrak dari publikasi peneliti tersebut dan Gambar 4.16 menunjukkan hasil transformasi *bag-of-words* menjadi *bag-of-phrase*. Pada kedua gambar tersebut, terlihat bahwa publikasi "Vincent Y. Lum" memiliki delapan *frequent phrase.* Dengan CTL Berbasis Frase, kata "*natural*" yang merupakan bagian dari frase "*natural language*" akan memiliki kemungkinan lebih besar untuk dikaitkan dengan topik "*text processing*" daripada kata "*natural*" yang di-*sampling* per kata. Dengan model topik yang lebih representatif, rekomendasi kolaborator yang diberikan akan lebih baik pula.

Visual Query Specification in a Multimedia Database System. In this paper we describe a visual interface for a Multimedia Database Management System (MDBMS). In spite of the technological advances in display devices, DBMS query languages are still linear in syntax as it was two decades ago. Although natural language interfaces have been found to be useful, natural language is ambiguous and difficult to process. For queries on standard (relational) data these difficulties may be easily avoided with the use of a visual, graphical interface to guide the user in specifying the query. For image and other media data which are ambiguous in nature, we use natural language processing combined with direct graphical access to the domain knowledge to interpret and evaluate the natural language query. The system fully supports graphical and image input/output in different formats. The combination of visual effect and natural language specification, the support of media data, and the allowance of incremental query specification are very effective to simplify the process of query specification not only for image or multimedia databases but also for all databases.

Gambar 4.15 Contoh Publikasi dengan Hasil CTLBF dan CTL Berbeda

visualization,query,specific,multimedia,databases,system,paper
describe,visualization,interface,multimedia,databases,**management system**,spite,technology,advanced,display,devices,dbms,query,language,linear,syntax,decade,ago,**natural language**,interface,found,**natural language**,ambiguity,difficult,process,query,standard,related,data,difficulties,easily,avoid,visualization,graphics,interface,guide,user,query,image,media,data,ambiguity,natural,**natural language**,process,combining,direct,graphics,access,**domain knowledge**,interpretation,evaluation,**natural language**,query,system,fully,support,graphics,**image input**,output,formation,combining,visualization,effective,**natural language**,specific,support,media,data,allowing,incremental,query,specific,effective,simplified,process,query,specific,image,multimedia,databases,databases

Gambar 4.16 Publikasi pada Gambar 4.15 dalam *Bag-Of-Phrases*

BAB 5
KESIMPULAN DAN SARAN

5.1. Kesimpulan

Kesimpulan yang dapat diambil berdasarkan hasil pengujian dan analisa terhadap metode yang diusulkan adalah sebagai berikut:

1. *Cross-Domain Topic Learning* Berbasis Frase menunjukkan nilai *precision* dan *recall* terbaik dengan nilai *minimum support* dalam proses *frequent phrase mining* sebesar 50.
2. Dengan jumlah publikasi sebanyak 3862 pada domain asal dan 2190 pada domain target, *Cross-Domain Topic Learning* Berbasis Frase menghasilkan nilai *precision* dan *recall* paling baik dengan konfigurasi jumlah topik pada domain asal dan target masing-masing sebesar 50.
3. Nilai *precision* dan *recall* yang rendah pada hasil rekomendasi baik CTL maupun CTL Berbasis Frase salah satunya disebabkan karena data latih untuk membentuk distribusi probabilitas yang baik kurang mencukupi.
4. *Cross-Domain Topic Learning* Berbasis Frase memiliki nilai *precision* lebih baik daripada rekomendasi kolaborasi penelitian dengan CTL berbasis *bag-of-words*, dengan peningkatan 9.55% pada hasil 10 rekomendasi teratas dan 5,1% pada 20 rekomendasi teratas. Hal ini disebabkan distribusi probabilitas peneliti terhadap topik dapat direpresentasikan dengan lebih baik, terutama pada data publikasi yang mengandung *frequent phrase.*

5.2. Saran

Saran yang dapat digunakan untuk pengembangan selanjutnya dari metode rekomendasi kolaborasi antardomain adalah:

1. Pemilihan dataset yang lebih tepat untuk memperoleh data kolaborasi penelitian yang cukup pada data latih dan data uji sehingga proses pemodelan topik dapat menghasilkan distribusi probabilitas keterkaitan peneliti terhadap topik yang lebih baik lagi.
2. Penggunaan fitur tambahan yang dapat memberikan informasi lebih banyak mengenai keterkaitan seorang peneliti dengan suatu topik, misalnya bibliografi, serta keterkaitan antarpeneliti, misalnya afiliasi peneliti.

[Halaman ini sengaja dikosongkan]

DAFTAR PUSTAKA

Abed, Mazin, Mohd Khanapi, Abd Ghani, and Raed Ibraheem. 2017. "Automatic Segmentation and Automatic Seed Point Selection of Nasopharyngeal Carcinoma from Microscopy Images Using Region Growing Based Approach." *Journal of Computational Science*. Elsevier B.V. https://doi.org/10.1016/j.jocs.2017.03.009.

Blei, David M., and John D. Lafferty. 2009. "Visualizing Topics with Multi-Word Expressions," 1–12. http://arxiv.org/abs/0907.1013.

Danilevsky, Marina, Chi Wang, Nihit Desai, Jingyi Guo, and Jiawei Han. 2013. "KERT: Automatic Extraction and Ranking of Topical Keyphrases from Content-Representative Document Titles." http://arxiv.org/abs/1306.0271.

El-Kishky, Ahmed, Yanglei Song, Chi Wang, Clare R. Voss, and Jiawei Han. 2014. "Scalable Topical Phrase Mining from Text Corpora." *Proceedings of the VLDB Endowment* 8.

Fujiwara, Yasuhiro, Makoto Nakatsuji, Makoto Onizuka, and Masaru Kitsuregawa. 2012. "Fast and Exact Top-K Search for Random Walk with Restart." In *Proceedings of the VLDB Endowment*, 442–53. https://doi.org/10.14778/2140436.2140441.

Han, Jiawei, and Chi Wang. 2014. "Mining Latent Entity Structures from Massive Unstructured and Interconnected Data." *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data - SIGMOD '14*. https://doi.org/10.1145/2588555.2588890.

Hemptinne, J De, J Ferrasse, A Gorak, S Kjelstrup, F Maréchal, O Baudouin, and R Gani. 2017. "Chemical Engineering Research and Design Energy Efficiency as an Example of Cross-Discipline." *Chemical Engineering Research and Design* 119. Institution of Chemical Engineers:183–87. https://doi.org/10.1016/j.cherd.2017.01.020.

Kang, Yong-Bin, Yuan-Fang Li, and Ross L. Coppel. 2015. "Capturing Researcher Expertise through MeSH Classification." *Proceedings of the Knowledge Capture Conference on ZZZ - K-CAP 2015*, 1–8. https://doi.org/10.1145/2815833.2815837.

Liang, Wei, Xiaokang Zhou, Suzhen Huang, Chunhua Hu, Xuesong Xu, and Qun Jin.

2017. “Modeling of Cross-Disciplinary Collaboration for Potential Field Discovery and Recommendation Based on Scholarly Big Data.” *Future Generation Computer Systems*. Elsevier B.V. https://doi.org/10.1016/j.future.2017.12.038.

Lindsey, Robert V., William P. Headden III, and Michael J. Stipicevic. 2012. “A Phrase-Discovering Topic Model Using Hierarchical Pitman-Yor Processes.” *Proceedings of the 2012 Joint Conerence on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, no. July:214–22.

Mitchell, Michael, Susan A Moore, Sarah Clement, Michael Lockwood, Gill Anderson, Suzie M Gaynor, Louise Gilfedder, Ross Rowe, Barbara Norman, and Edward C Lefroy. 2017. “Biodiversity on the Brink : Evaluating a Transdisciplinary Research Collaboration.” *Journal for Nature Conservation* 40 (December 2016). Elsevier:1–11. https://doi.org/10.1016/j.jnc.2017.08.002.

Osuna, Francisco, Monika Akbar, and Ann Q. Gates. 2017. “On Using Disparate Scholarly Data to Identify Potential Members for Interdisciplinary Research Groups.” *2017 IEEE International Conference on Information Reuse and Integration (IRI)*, 59–68. https://doi.org/10.1109/IRI.2017.33.

Pay, Tayfun. 2016. “Totally Automated Keyword Extraction.” *Proceedings - 2016 IEEE International Conference on Big Data, Big Data 2016*, 3859–63. https://doi.org/10.1109/BigData.2016.7841059.

Schone, Patrick, and Daniel Jurafsky. 2001. “Is Knowledge-Free Induction of Multiword Unit Dictionary Headwords a Solved Problem?” *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*, 100–108.

Tang, Jie, Sen Wu, Jimeng Sun, and Hang Su. 2012. “Cross-Domain Collaboration Recommendation.” *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD ’12*, 1285. https://doi.org/10.1145/2339530.2339730.

Wang, Qi, Jian Ma, Xiuwu Liao, and Wei Du. 2017. “A Context-Aware Researcher Recommendation System for University-Industry Collaboration on R & D Projects.” *Decision Support Systems* 103. Elsevier B.V.:46–57. https://doi.org/10.1016/j.dss.2017.09.001.

Wang, Xuerui, Andrew McCallum, and Xing Wei. 2007. “Topical N-Grams: Phrase and Topic Discovery, with an Application to Information Retrieval.” *Proceedings - IEEE International Conference on Data Mining, ICDM*, 697–702. https://doi.org/10.1109/ICDM.2007.86.

BIOGRAFI PENULIS

Vit Zuraida lahir di Probolinggo, Jawa Timur pada tanggal 9 Januari 1989. Pendidikan yang telah ditempuh adalah SD, SMP, dan SMA Taruna Dra. Zulaeha dan Ilmu Komputer Universitas Indonesia (2017-2011). Rumpun Mata Kuliah (RMK) yang diambil oleh penulis adalah Komputasi Cerdas dan Visi (KCV). Alat komunikasi yang disediakan oleh penulis adalah vit.zuraida@gmail.com